Audio-Visual Fusion: New Methods and Applications

THÈSE N^{O} 4962 (2011)

PRÉSENTÉE LE 25 FÉVRIER 2011

À LA FACULTÉ SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE DE TRAITEMENT DES SIGNAUX 2

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Anna LLAGOSTERA CASANOVAS

acceptée sur proposition du jury:

Dr J.-M. Vesin, président du jury
Prof. P. Vandergheynst, directeur de thèse
Prof. F. Marqués, rapporteur
Dr G. Monaci, rapporteur
Prof. J.-Ph. Thiran, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2011

Look up to the sky
You'll never find rainbows
If you're looking down.

– Charlie Chaplin –

Acknowledgments

During the four years of my PhD, I had the chance to meet and collaborate with several valuable colleagues and friends that have supported me and have made this work possible. I would like to express my gratitude to all of them.

The first person I want to thank is my advisor, Pierre Vandergheynst. Thank you for giving me the opportunity to work on a very interesting subject, for all the discussions and for the freedom you gave me during my thesis.

I would like to thank the members of my thesis jury: Dr. Jean-Marc Vesin, Prof. Ferran Marqués, Dr. Gianluca Monaci and Prof. Jean-Philippe Thiran. Thank you for your careful reading of this manuscript and your helpful comments and suggestions.

Part of the work present in this dissertation has been done in collaboration with colleagues and students working under my supervision. Working with them has been very important for my research and personal development. First, I would like to thank Gianluca Monaci for giving me the opportunity to do my Master Thesis at LTS2 and guiding and supporting me during the first year of PhD. I would also thank Rémi Gribonval for the great time spent in Rennes and for having revealed me the secrets of sound source separation. Finally, I would like to thank Patricia Calatayud for her excellent work during her master thesis.

A special thank goes to all the members of the Signal Processing Laboratories where I have had the opportunity to work in a wonderful, stimulating environment. These years in Lausanne would have not been the same without the baby foot competitions, discofit sessions, Barça games with tapas at Centro Asturiano, board games at Elda's place and Friday nights at SAT!

I would also like to thank my office mates for their support. Thank you Matteo for reading a part of this manuscript and Florian for being there during the last months and sharing the stress load.

Finally, thank to all the persons who gave me support during the last part of my thesis by reading and correcting this manuscript, paying attention during the practice for the defense or simply cheering me up in difficult moments. Special thanks to Gilles for his help with the French translation of the abstract!

També voldria agrair a la meva família el seu suport incondicional. Gràcies per haver-me animat a venir a Lausanne i per rebre'm amb els braços oberts cada vegada que torno a casa. Amb totes les vostres visites i trucades sembla que us tingui més aprop.

I moltes gràcies Lluís per haver estat sempre al meu costat i per animar-me a tirar endavant quan ja no podia més. Aquests tres anys vivint junts a Lausanne han estat immillorables!

Table of contents

Abstract

The perception that we have about the world is influenced by elements of diverse nature. Indeed humans tend to integrate information coming from different sensory modalities to better understand their environment. Following this observation, scientists have been trying to combine different research domains. In particular, in joint audio-visual signal processing the information recorded with one or more video-cameras and one or more microphones is combined in order to extract more knowledge about a given scene than when analyzing each modality separately.

In this thesis we attempt the fusion of audio and video modalities when considering *one* video-camera and *one* microphone. This is the most common configuration in electronic devices such as laptops and cellphones, and it does not require controlled environments such as previously prepared meeting rooms. Even though numerous approaches have been proposed in the last decade, the fusion of audio and video modalities is still an open problem. All the methods in this domain are based on an assumption of synchrony between related events in audio and video channels, i.e. the appearance of a sound is approximately synchronous with the movement of the image structure that has generated it. However, most approaches do not exploit the spatio-temporal consistency that characterizes video signals and, as a result, they assess the synchrony between single pixels and the soundtrack. The results that they obtain are thus sensitive to noise and the coherence between neighboring pixels is not ensured.

This thesis presents two novel audio-visual fusion methods which follow completely different strategies to evaluate the synchrony between moving image *structures* and sounds. Each fusion method is successfully demonstrated on a different application in this domain.

Our first audio-visual fusion approach is focused on the modeling of audio and video signals. We propose to decompose each modality into a small set of functions representing the structures that are inherent in the signals. The audio signal is decomposed into a set of atoms representing concentrations of energy in the spectrogram (sounds) and the video signal is concisely represented by a set of image structures evolving through time, i.e. changing their location, size or orientation. As a result, meaningful features can be easily defined for each modality, as the presence of a sound and the movement of a salient image structure. Finally, the fusion step simply evaluates the co-occurence of these *relevant events*. This approach is applied to the blind detection and separation of the audio-visual sources that are present in a scene.

In contrast, the second method that we propose uses basic features and it is more focused on the fusion strategy that combines them. This approach is based on a nonlinear diffusion procedure that progressively erodes a video sequence and converts it into an *audio-visual* video sequence, where only the information that is required in applications in the joint audio-visual domain is kept. For this purpose we define a diffusion coefficient that depends on the synchrony between video motion

and audio energy and preserves regions moving coherently with the presence of sounds. Thus, the regions that are least diffused are likely to be part of the video modality of the audio-visual source, and the application of this fusion method to the unsupervised extraction of audio-visual objects is straightforward.

Unlike many methods in this domain which are specific to speakers, the fusion methods that we present in this thesis are completely general and they can be applied to all kind of audio-visual sources. Furthermore, our analysis is not limited to one source at a time, i.e. all applications can deal with multiple simultaneous sources. Finally, this thesis tackles the audio-visual fusion problem from a novel perspective, by proposing creative fusion methods and techniques borrowed from other domains such as the blind source separation, nonlinear diffusion based on partial differential equations (PDE) and graph cut segmentation.

Keywords

Audio-visual signal processing, sparse representation, redundant dictionary, blind source separation, audio-visual source, nonlinear video diffusion, graph cut segmentation.

Résumé

Notre perception du monde est influencée par de nombreux éléments nous entourant. En effet, nous utilisons constamment l'information perçue par chacun de nos sens afin de mieux comprendre notre environnement. S'inspirant de cette idée, les scientifiques ont donc essayé de combiner différents domaines de recherche. En particulier, en traitement du signal audio-visuel, l'information enregistrée avec plusieurs microphones et caméras est traitée de manière conjointe plutôt que séparément afin d'améliorer la compréhension d'une scène.

Dans cette thèse, nous combinons l'information audio-visuelle d'un seul microphone et d'une seule caméra. Cette configuration est très usuelle sur de nombreux appareils électroniques (téléphones portables, ordinateurs, etc.) et ne nécessite pas d'environnement contrôlé (comme une salle de conférence spécialement préparée). Bien que de nombreuses recherches aient été faites dans ce domaine ces dernières années, la combinaison de l'information audio et vidéo est toujours un problème d'actualité. Toutes les méthodes développées sont basées sur une hypothèse de synchronisme entre les événements sur les pistes audio et vidéo. En effet, un son est toujours plus ou moins synchronisé avec le mouvement qui l'a produit. Cependant la plupart des méthodes n'exploitent pas la cohérence spatio-temporelle qui caractérise le signal vidéo. Par conséquent, ces techniques n'estiment la synchronisation qu'entre le son et des pixels isolés. Les résultats sont donc sensibles au bruit et n'assurent pas de cohérence entre pixels voisins.

Cette thèse présente deux méthodes audio-visuelles qui suivent de nouvelles stratégies pour évaluer le synchronisme entre des *régions* en mouvement et le son. Chaque méthode est testée pour une application spécifique, et obtient des résultats satisfaisants.

La première méthode est d'abord basée sur une modélisation séparée des signaux audio et vidéo. Nous proposons de décomposer chacun de ces signaux en plusieurs ensembles de fonctions représentant leurs structures internes. Ainsi, le signal audio est décomposé par un ensemble d'atomes de référence représentant la concentration d'énergie dans son spectrogramme. Le signal vidéo est lui représenté par un ensemble de structures géométriques évoluant dans le temps en changeant de taille, de position ou d'orientation. Les principales caractéristiques des signaux audio ou vidéo, comme la simple présence d'un son ou d'un mouvement, peuvent donc facilement être identifiées. La dernière étape de cette méthode consiste à combiner les informations caractéristiques extraites des signaux audio à celles des signaux vidéo afin d'identifier la cooccurrence d'évènements. Cette technique est testée dans le cadre de la détection et la séparation aveugle audio-visuelle de sources présentes dans une scène.

La seconde méthode proposée utilise une représentation plus simple des signaux audio et vidéo et est plus concentrée sur la combinaison de l'information entre ces deux signaux. Elle est basée sur une diffusion non-linéaire qui modifie progressivement la séquence vidéo en tenant compte de la

séquence audio. Ce processus aboutit à une séquence audio-visuelle où seule l'information nécessaire à nos applications est gardée. Dans ce but, nous définissons un coefficient de diffusion dépendant du synchronisme entre le mouvement dans la vidéo et l'énergie audio. Ce coefficient met en évidence les régions en mouvement, corrélées avec la présence de son. Par conséquent, les régions où la diffusion est la plus faible indiquent, très probablement, l'emplacement d'une source audio-visuelle. L'application de cette technique est testée pour l'extraction non supervisée des sources sur la bande vidéo.

A l'inverse des nombreuses méthodes spécifiques à la parole dans ce domaine, les méthodes développées dans cette thèse sont générales et peuvent s'étendre à n'importe quels types de sources audio-visuelles. De plus, elles ne sont pas limitées à une seule source mais s'appliquent à de multiples sources simultanées. Enfin, cette thèse résout le problème de la combinaison audio-visuelle par de nouvelles approches empruntées à d'autres domaines comme la séparation aveugle de source, la diffusion non linéaire par équations aux dérivées partielles (EDP) et la segmentation par coupure de graphe.

Liste des mots-clefs

Traitement du signal audio-visuel,
dictionnaire redondant,
source audio-visuelle,
segmentation par coupure de graphe.
représentations parcimonieuses
séparation aveugle de source
diffusion non linéaire

List of figures

List of tables

Glossary

α	Audio-visual diffusion ratio
$a(t)$	Audio signal
$A(t, f)$	Short Time Fourier Transform (STFT) of the audio signal
c	Audio-visual coherence
χ	Correlation score between audio and video atoms
$\mathcal{D}^{(m)}$	Redundant dictionary of unit norm atoms for modality m
D	Diffusion coefficient
$\mathfrak{f}^{(m)}$	Feature corresponding to an atom in modality m
$I(x, y)$	Image frame
l_p	Binary label assigned to pixel p
Λ^{color}	GMM characterizing the color statistics of a source
Λ^{spec}	Spectral GMM characterizing the acoustic frequency behavior of a source
$\mathcal{P}^j$	Set of pixels in the j-th GoF
$\phi^{(m)}$	Normalized atom for modality m
S	Audio-visual source
s_σ	Regularized audio-video synchrony
T	Segmentation trimap
τ	Diffusion time
$v(x, y, t)$	Video signal
w	Source's activity vector estimated through joint audio-visual processing
$\mathbf{x}$	3D video coordinates

1 Introduction

1.1 Motivation

The perception that we have about the world is influenced by elements of diverse nature. Indeed humans tend to integrate information coming from different sensory modalities to better understand their environment. For example, when thinking about a fruit, we can remember its color and shape, but also its smell, taste and texture. In the audio-visual domain, several studies show that the listener can exploit the correspondence between speaker lips movements and the produced sounds to better understand speech, especially in adverse environments [24, 78, 79]. The speech recognition task is thus facilitated by the integration of acoustic and visual stimuli.

Following this observation, scientists have been trying to combine different research domains. In joint audio-visual analysis, nowadays it is possible to use the video information to improve results in the audio domain for applications such as speech recognition [49, 65], speech enhancement [22, 29, 30] and sound source separation [20, 66, 67, 77, 84]. Other methods try to assess coherence between both modalities to track [63] or locate sound sources in the video signal [26, 36, 58, 74–76]. Some latter approaches go one step beyond and try to separate the scene into audio-visual structures, each of them composed by a visual part and the associated soundtrack [6, 45, 73]. All these applications can then be used for automatic management of videoconferences, indexing and segmentation of multimedia data [38, 69], and robotics [27]. Some other recent applications that combine information in audio and video modalities are automatic speaker recognition [18] for biometric person authentication [9, 17], emotion recognition [86], automatic music transcription [28] and video classification [34].

Let us now discuss the main reasons to focus this research on the fusion of audio and video signals.

- Even though in recent years many electronic devices such as laptops, cellphones and even video game consoles have integrated video cameras and microphones to their hardware, relatively few approaches explore the possibilities of combining audio and video modalities in creative

applications.

- The fusion of audio and video modalities is still an open problem. Most approaches in this domain first define features for each modality, and then they use these representations in a fusion step, which tries to assess the synchrony between modalities using statistical tools such as canonical correlation analysis or joint audio-visual probabilities for the features. However, there is not a standard method for the audio-visual fusion. Furthermore, most of those approaches assess the synchrony between the behavior of single pixels and the soundtrack, instead of considering the movements of image *structures*. As a result, they do not exploit the spatial coherence of video signals and that makes them vulnerable to visual noise.

- Audio-visual signals represent the subset of multimodal signals that has attracted the interest of more researchers. Multimodal signals are signals captured by two or more different kinds of sensors (devices) that are observing the same scene. Then, the multimodal signal processing attempts the effective fusion of the information present in each modality. For example, in medical imaging the information from magnetic resonances (MR) and computed tomography (CT) scans is combined for registration and segmentation purposes [13, 50].

1.2 Joint Audio-Visual Signal Processing

Audio and video modalities capture different information of the same scene. The video signal contains the information about the appearance (color, texture, shape) and distribution of the objects in the scene, while the sounds (speech, music, noise) are only available in the audio signal. As discussed before, humans combine in a natural way the information in audio and video modalities. For example, we can easily understand the relationship between an object that is falling and the sound of the crash, we intuitively link moving lips to the presence of speech, and we know the kind of music that we will hear when we see a guitarist's arm moving. Thus, through the joint processing of audio and video signals we can better understand a scene than when considering each modality separately. Again in the human case, we can use lip-reading to detect the speaker between two persons that move the lips, and it is possible to assign the sounds to the corresponding music instrument when we are in a concert.

In general, audio-visual fusion methods integrate the information that is present in the video signal captured with one or more video-cameras and the audio signal recorded with one or more microphones. When considering the video domain, two or more video cameras allow a 3D understanding of the observed scene, where a depth can be associated to each object [37, 70–72]. Regarding the audio domain, microphone arrays are commonly used to localize and separate the sound sources in the scene [2, 59, 87]. Several examples of audio-visual fusion approaches using multiple video-cameras and/or microphones can be found in [7, 35, 63]. However, those configurations (with several audio or video sensors) need some calibration and they can not be applied to general situations, but rather to controlled environments such as previously prepared meeting rooms. Even though many electronic devices integrate video cameras and microphones to their hardware, in most cases only one sensor of each modality is available. Notice that two microphones are present in some devices, but they are located so close that the recorded audio signals are very similar. Here we consider the simplest but also the most common audio-visual configuration, where the content of the scene is captured by *one* microphone and *one* video-camera.

Figure 1.1 depicts thus the typical baseline that we consider in this thesis. We have a three-dimensional video signal recorded with a video-camera and the corresponding one-dimensional audio

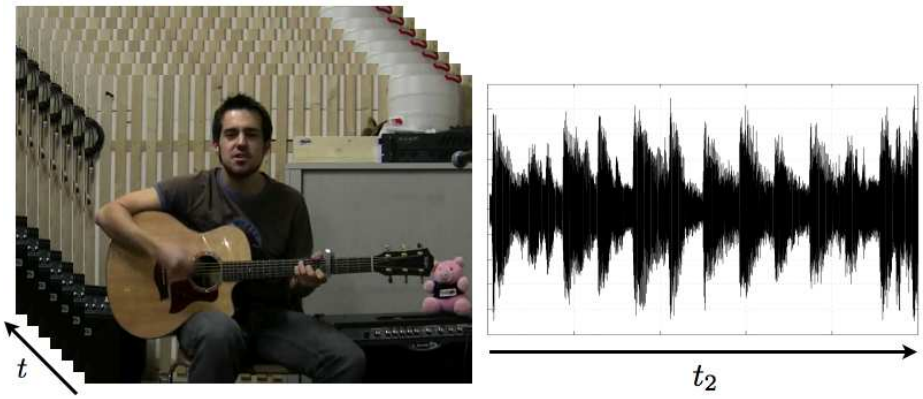


Figure 1.1 – *Example of a 3D video signal [left] and the corresponding 1D audio signal [right]. The temporal axis of each modality presents a different resolution.*

signal coming from one microphone. Audio and video signals share a temporal axis, but the resolution of this axis is different. Typically, we have much more audio samples than video frames since the sampling rate of the audio signal is much higher. Then the challenge lies in efficiently combining the information in both channels in order to extract a maximum of knowledge about the scene that we observe.

1.2.1 Basic Assumption

The information present in audio and video modalities has a very diverse nature. Furthermore, audio and video signals have different dimensionality and temporal resolution. Thus, some assumptions need to be made in order to combine both modalities.

Several works in audio-visual perception have demonstrated the correlation between audio and video modalities in the speech case [24, 78, 79]. Specifically, they showed that the correspondence between the speaker lips movements and the produced sounds can be exploited by the listener to better understand speech, especially in noisy environments. This is particularly evident when we think about lip-reading, i.e. the technique of understanding speech by visually interpreting the movements of the lips (and also the face and tongue in a minor degree). Another example that demonstrates the relationship between hearing and vision in speech perception is the McGurk effect described in [52]. This effect may be experienced when we combine a video of a person uttering one phoneme with a soundtrack corresponding to a different phoneme. In this case, the perceived phoneme is often an intermediate phoneme different from the video and the audio ones. For example, a visual /ga/ combined with an audio /ba/ can be heard as /da/ because of the effect associated to the video interference.

In this thesis, we do not restrict the analyzed audio-visual sequences to movies containing speakers. As a result, we need to base our fusion methods on an assumption that applies to all kind of audio-visual sources. The assumption that we use in this thesis is common in all applications in the joint audio-visual processing domain. It states that the presence of a sound is *approximately* synchronous with the movement that has generated it. Thus, related events in audio and video channels happen at approximately the same time (small lags can appear due to the different arrival times for each sensor). Several studies on audio-visual perception even support the idea that when there is a small temporal shift between events in audio and video modalities, the brain tends to

perceive them as being synchronous [25, 82]. Music instruments represent some other examples of the synchrony between motion and sounds. The fingers movements are coherent with the piano sounds, the rhythm is controlled by the periodicity with which the drumsticks hit the drums, and the hands movements are also correlated with the guitar sounds. In fact, audio and video modalities are observing the same scene and they only share the temporal axis (see Figure 1.1). Thus, synchrony is the only way to link both channels if we do not have previous knowledge about the characteristics of the sources in the scene.

Let us now discuss the main challenges in the joint processing of audio and video signals according to the assumption of synchrony between related events. A first challenge in this domain is to distinguish the distracting motion from the motion related to the sounds, since the distracting motion can also be sporadically synchronous with the soundtrack. Another important challenge is the presence of multiple audio-visual sources: in this case some sounds are related to some movements in a part of the image while other sounds are synchronous with the motion in other locations. Furthermore, not all the sounds are generated by moving objects, i.e. a hi-fi equipment playing music represents a clear example of audio-only source. As a result, complex backgrounds composed of several moving objects and/or acoustic noises difficult the audio-visual fusion task when only one video-camera and one microphone are available. Finally, other challenges are common in most signal processing applications and are given by the video camera limitations (low quality, resolution and frame rate) and the microphone limitations (such as directivity patterns, internal noise and wind noise).

1.2.2 Audio-Visual Fusion Methods: State of the Art

In this section we explain some of the most relevant approaches in the joint processing of audio and video signals when considering *one* microphone and *one* video-camera.

The first work in this domain was performed by **Hershey and Movellan** in 1999 [33]. Following the physiological evidence that the sounds spatial localization is influenced by their synchrony with the video signal (ventriloquism effect), they presented a method to locate sound sources in an image by joint audio-visual processing. For this purpose, they first defined features corresponding to audio and video signals and then they used them to assess the synchrony between modalities. In this work, the synchrony was defined as the degree of mutual information (MI) between the audio energy and each pixels' intensity evolution, and it was computed by means of the Pearson correlation coefficient [1]. Thus, pixels were treated independently and the speaker's position in a time window was estimated as the center of mass of the audio-visual mutual information. In this approach, the joint statistics of audio and video signals were assumed to be Gaussian, and pixels were assumed to be independent conditioned on the audio signal.

Most of the following approaches kept this setup: in a first stage they define features for audio and video signals separately and then they combine these features in a fusion step, which tries to assess the synchrony between modalities. Next, we explain the most representative methods that follow this strategy by grouping them in terms of the fusion technique that they use.

Estimate the joint probability density function (pdf) of audio and video features

This is probably the most common strategy when trying to combine the information in audio and video channels. Following the **Hershey and Movellan** preliminary work in this domain [33], several approaches used the mutual information (MI) to assess the synchrony between audio and video modalities.

Nock et al. tested in [58] (2003) three different approaches: the mutual information, assuming either joint Gaussian distributions or discrete distributions, and a specific measure adequate to the speaker case, which was based on the modelling of audio-visual features using Hidden Markov Models (HMM). This work represents the first exhaustive evaluation of the performance of several audio and video representations and fusion methods in this domain. The representations that they tested were the discrete cosine transform (DCT), the pixel's intensity and pixel's intensity changes for the video signal, and the mel-frequency cepstral coefficients (MFCC) for the audio signal. The analyzed sequences belong to the CUAVE audio-visual database in [62] and consist on two persons speaking in turns. The authors concluded that the pixel's intensity changes together with the Gaussian MI were the more suitable combination in the speaker localization task.

After this work, several information theoretic approaches attempted the fusion of audio and video modalities by maximizing the MI while removing the assumption that the joint distribution of audio and video features is Gaussian.

In [26] (2004), **Fisher and Darrell** developed a statistical measure to decide if a pair of audio and video signals come from a common source. First, they proposed a probabilistic generative model for audio-visual sequences. Then, they learnt the projections that needed to be applied to audio and video features in order to maximize the MI between modalities. The features that they use are images (pixel's intensity) and audio periodograms, and no training is required. However, the use of the Parzen density estimator in the MI computation introduces several parameters to tune. This approach is applied to the speaker localization task in the presence of background motion.

Butz and Thiran proposed in [13] (2005) a similar approach to determine the region in a video signal containing the speaker's mouth. In this work, audio and video signals were modeled using Markov chains. Then, the objective was to find the features in each modality that minimized the lower bounds on the error probabilities of the Markov chains. For this purpose, they maximized the feature efficiency coefficient, i.e. the ratio between the MI and the joint entropy of audio and video features. Thus, they try to find a solution in which the amount of information of each modality that is not related to the other modality is minimal. The features in this case are the pixels' intensity changes and the linear combination of the audio power spectrum carrying a maximum of entropy. This framework was also used by **Besson et al.** in [10] (2008) to locate the active speaker from several candidate mouth regions, which were extracted either manually or using a face detector. In this case, the video features correspond to the vertical component of the optical flow in the mouth, and the audio features are optimized by choosing the linear combination of the MFCC that maximizes the MI between audio and video.

In [32] (2006), **Gurban and Thiran** proposed to model the joint probability density function of audio and video features by means of a Gaussian mixture model (GMM). GMMs are able to approximate any density function provided that enough Gaussians are considered. In a training step they learn this joint distribution using expectation maximization on video signals containing only mouth regions. Then the speaker's location is estimated by finding the region in a video signal whose features present the maximum likelihood of being generated by this distribution. The features that they use are the logarithm of the audio energy and the difference between the optical flow in the top and bottom parts of the mouth.

Canonical Correlation Analysis (CCA)

In [75] (2000), **Slaney and Covell** presented a method to assess the degree of synchrony

between video facial images and speech. In a training stage, they used CCA to find the linear mapping maximizing the cross-correlation between the standard deviation from an aligned face image and different audio features, such as the MFCC and the audio energy. Then, this linear transform was used to map both modalities in a common space and evaluate their correlation as a function of time. CCA allows the comparison of two vectors of different nature and dimensions. As a result, this approach can assess the synchrony between the whole face (in a video frame) and the speech, by combining the information from all the pixels instead of evaluating the correlation of each pixel separately. Thus, the implicit assumption that pixels are independent conditioned on the audio signal is removed.

Kidron et al. overcame in [36] (2007) the need of a training step when using CCA. They solved the problem of having insufficient data in the estimation of the signals statistics by imposing an assumption of sparsity in the localization results, i.e. the authors assume that only a small number of video features are associated with the audio feature (energy) in a time window. A wavelet transform is applied to the temporal-difference images to extract the video features: only a few coefficients are required to represent the motion in the image plane. This method can be applied to all types of audio-visual sources (not only speakers) and it provides good results in the presence of non-stationary (moving) sources, distracting video motion and audio noise. However, its formulation does not allow the localization of multiple simultaneous audio-visual events, e.g. only one speaker would be detected if two persons speak at the same time.

Neural networks

Cutler and Davis presented in [19] (2000) a method to automatically locate the speaker in a video signal by using neural networks. The features in this case are the MFCC for the audio signal and a normalized correlation between consecutive frames (normalized intensity changes) for the video signal. Their procedure is divided into two steps. First, they train a time-delayed neural network (TDNN) on audio and video features corresponding to mouth regions during speech in order to learn audio-visual correlations. In the second step the speaker's mouth is located by choosing the spatio-temporal region that maximizes the TDNN output. This method is demonstrated on one sequence presenting a single speaker. The main limitations of this approach are the difficulty to control which characteristics the TDNN is learning and the need for a similarity measure invariant to the translations and rotations of the head and mouth region during speech.

Onsets co-ocurrence

In [6] (2007), **Barzelay and Schechner** proposed a method to identify the number of audio-visual sources in a scene by localizing the video structures more correlated to the soundtrack. For this purpose, they evaluate the synchrony between audio and video *onsets*, which are defined respectively as the beginning of a sound and a significant change on the speed or direction of a video structure (edge or corner). In this approach, the fusion step is computed by means of a matching criterion that favors coincidences between audio and video onsets and penalizes mismatches. This method is demonstrated on sequences presenting speakers and music instruments.

A different strategy consists on extracting the meaningful audio-visual structures that compose real video sequences. Once this information is obtained, it can be applied to the localization of

the sound sources in a scene. Here we detail two methods that follow this idea using very different techniques.

Principal Component Analysis (PCA) followed by Independent Component Analysis (ICA)

In [76] (2003), **Smaragdis and Casey** proposed a method to extract audio-visual independent components from sequences containing semi-static objects. The procedure was performed on a fused data set (audio and video features were concatenated) and it was divided into two steps: they first performed a dimensionality reduction by means of principal component analysis (PCA) to keep the dimensions with maximal variance, and then they ensured the maximal independence of the result by using independent component analysis (ICA). After this procedure, the scene is decomposed into a set of audio-visual functions (component bases) presenting different activations through time (component weights). The audio and video features in this case are respectively the magnitude of the audio spectrum and the pixel's intensity. The main limitation of this approach is that it can not deal with dynamic scenes: objects with an important motion are represented by several component bases instead of composing a unique audio-visual function.

Dictionary Learning

Monaci et al. proposed in [55] (2007) a method to learn the multimodal structures that are recurrent in audio-visual sequences. Once this step is achieved, it is possible to build a redundant multimodal dictionary, which is composed of all the possible translations of this audio-visual basic functions. The learning algorithm that they propose enforces synchrony between modalities and de-correlation between the (shift invariant) basis functions in the dictionary. In this work, the speaker localization task is performed by finding the positions of maximal projection of the learned functions in the video signal, and grouping together the positions in the image plane by using a clustering algorithm. Results in the presence of distracting video motion and acoustic noise are encouraging.

Finally, the approach proposed by **Monaci et al.** in [53] (2006) deserves a special mention since it is the basis of the first audio-visual fusion method that we present in this thesis. This approach focuses on the understanding and corresponding modelling of audio and video modalities. In a first stage, the video signal is decomposed into a small set of 3D structures representing the geometric image components (edges) and the temporal transformations that they experiment. Thus, a small number of functions concentrates concisely most of the information in the video modality. Then, the sound source localization task is limited to the search of the 3D structure whose motion better fits the energy variations in the audio channel, which is denoised by means of the matching pursuit algorithm (MP) [51]. The video representation is so powerful that a simple scalar product is enough to assess the synchrony between audio and video features. Good results are obtained in the spatial localization of *general* sound sources. However, this approach is not designed to deal with the localization of multiple simultaneous audio-visual events, i.e. two or more audio-visual sources active at the same time can not be detected. The main differences between this method and our first audio-visual fusion approach are explained in depth in Chapter 2.

Table 1.1 summarizes the main characteristics of the audio-visual fusion methods that we have explained in this section. Most of the approaches in this domain first define features for each modality such as the audio energy [33, 36, 53] or mel-frequency cepstral coefficients (MFCC) [10, 19, 58, 75] for the audio, and pixel intensities [13, 33, 76] or temporal variations [19, 36, 58] for the video. Then, they use these representations in a fusion step, whose objective is to assess the synchrony

METHOD	A FEATURE	V FEATURE	AV PDF	FUSION	TRAINING	GENERAL
[33]	energy	pixels' intensity	Gaussian	max MI	-	-
[58]	MFCC	pixels' intensity pixels' changes DCT	Gaussian discrete HMM	max MI	- - ✔	-
[26]	periodogram	whitened video	general	max MI	-	-
[13]	spectrum	pixels' intensity	general	$\max e(A,V)$	-	-
[10]	MFCC	optical flow	general	$\max e(A,V)$	-	-
[32]	log energy	optical flow	GMM	max likelihood	✔	-
[75]	MFCC	image changes	-	CCA	✔	-
[36]	energy	wavelet coeffs.	-	CCA+sparsity	-	✔
[19]	MFCC	image changes	-	TDNN	✔	-
[6]	onsets	structure onsets	-	matching criterion	-	✔
[76]	spectrum	image	-	PCA+ICA	-	✔
[55]	samples	whitened video	-	dictionary learning	✔	-
[53]	energy	structures' motion	-	scalar product	-	✔

Table 1.1 – *Summary of the main characteristics of the audio-visual fusion methods reviewed in this section. For each method we depict [from left to right] the audio and video features that they use, the joint probability density function (pdf) of the features that they try to estimate (in the case they do so), the techniques that they use in the fusion step, if an off-line training procedure is required and, finally, if they are applied to general sound sources or exclusively to the speaker case. In the "fusion" column, the value* $\max e(A,V)$ *corresponds to maximizing the* feature efficiency coefficient*, i.e. ratio between the MI and the joint entropy of audio and video features.*

between both modalities using canonical correlation analysis [36, 75] or through the estimation of the joint densities of audio and video features [10, 13, 26, 32, 33, 58]. Many methods evaluate the audio-visual synchrony for each pixel independently [10, 13, 33, 58], and thus make the implicit assumption that pixels are independent conditioned on the audio signal. In most cases the audio-visual fusion methods are designed for dealing with speakers. In [10, 19, 32, 75] for example the video features are extracted from the face/mouth region and, as a result, these methods could not be applied to general sound sources. As a final remark, many approaches do not even consider the case in which several audio-visual sources (or speakers) are active at the same time. Their objective is to localize *the* sound source, i.e. they look for a maximum of synchrony.

1.3 Thesis Contributions

This thesis tries to overcome the limitations of previous methods in the joint processing of audio and video signals. For this purpose, we propose two innovative audio-visual fusion methods, each of them demonstrated on a different application in this domain.

1. In the first approach, audio and video modalities are decomposed into a small number of functions taken from redundant dictionaries of signals. These basic functions represent respectively concentrations of audio energy in the spectrogram and salient image structures and their temporal evolution. Then, the audio-visual fusion step consists on quantifying (through a scalar product) the synchrony between relevant events in audio and video modalities, which are defined as the presence of a sound and the movement of an image structure.
   This approach is applied to *blind* audio-visual source separation: given a scene in which several audio-visual sources are present, our method is able to extract a movie containing each source

separately. For example, when analyzing a mixture of two speakers our objective is to obtain two separated audio-visual signals, each of them containing the face and corresponding voice of one speaker, without the interference generated by the other one. This is a very challenging task, specially when two sources are active simultaneously and the soundtrack needs to be separated.

2. The second method concentrates all audio-visual knowledge on a nonlinear diffusion procedure which will iteratively modify the video signal and convert it into an *audio-visual* video sequence, that is a sequence containing only the essential information for joint audio-visual processing. In this case, the fusion between modalities is performed implicitly by building an audio-visual diffusion coefficient which depends on an estimate of the synchrony between sounds and motion. Video structures contributing to the soundtrack are preserved while the remaining information is iteratively removed. The spatio-temporal characteristic of the proposed 3D diffusion process ensures coherence between neighboring pixels. As a result, this second method evaluates also the synchrony between the sounds and moving *regions* (pixels are not considered independently).
   The proposed audio-visual nonlinear diffusion approach is applied to the unsupervised detection and segmentation of the audio-visual objects in a scene, i.e. the video modality of the audio-visual sources is automatically extracted. While most approaches in this domain only attempt the spatial localization of the source, our objective is to extract the object (e.g. a face or a hand) whose motion generates the sounds. For this purpose we use a modified graph cut segmentation procedure that takes into account both the original video signal and the *audio-visual* video sequence that we obtained through diffusion.

To summarize, in this thesis we present two audio-visual fusion methods that follow completely different strategies. The first approach focuses on the modeling of the signals and, as a result, the fusion method does not need to be very complex. In contrast, the second method uses very basic features (video signal and audio energy) which are combined by means of a powerful fusion technique that ensures spatio-temporal coherence.

Let us now detail the main differences between the previous methods reviewed in Section 1.2.2, and the two audio-visual fusion approaches in this thesis.

- The fusion methods that we propose are based on two main assumptions. The first one was explained in Subsection 1.2.1. This assumption is common in all the approaches in this domain and it states that the presence of a sound and the movement that has generated it are approximately *synchronous*. The second one is the assumption that most methods neglect, and it is based on the knowledge that video signals are composed of a set of homogeneous regions (image structures) whose characteristics evolve through time. Thus, *neighboring pixels* (both in space and time) *are often highly correlated*. In contrast, most of the previous approaches [13, 26, 33, 58] consider them independently. This characteristic makes them more vulnerable to visual noise and it does not ensure the spatial consistency of the result. Even though the fusion methods that we present in this thesis are completely different, they both assess the synchrony between moving *regions* (image structures) and the soundtrack, i.e. the spatio-temporal coherence that is characteristic from video signals is always taken into consideration.

- In this thesis we attempt the fusion of audio and video modalities in general sequences. Unlike many of the previous approaches [13, 19, 26, 33, 58, 75], our analysis is not focused on speakers or talking faces. The two audio-visual fusion methods that we propose are completely general and thus they can handle all kind of audio-visual sources. In fact, our fusion methods are

demonstrated in sequences containing speakers and music instruments. Our approaches are based on two assumptions which are reliable in all situations: the synchrony between audio and video channels and the knowledge that video signals are composed by homogeneous regions separated by edges.

- Some of the fusion methods reviewed in Section 1.2.2 require the previous extraction of regions that are (or can be) related to the soundtrack (e.g. face or mouth region) either for the training [19, 32, 75] or for the testing [10, 26, 75]. In contrast, such a preprocessing is not needed for the two fusion methods that we present, since they can deal with general data. Furthermore, we do not require any off-line training stage.

- Most of the previous approaches can not handle multiple simultaneous sources, i.e. the presence of two or more audio-visual sources that are active at the same time. Those approaches try to localize *the* source in a time period without even considering that in general situations there *can* possibly be several sources [26, 33, 36, 53, 58]. For example, in a meeting room it is common that several persons speak at the same time, and music instruments are mixed most of the time in a concert. The fusion methods that we show in this thesis are very general and, as a result, all our approaches and applications can deal with multiple simultaneous audio-visual sources.

- In this thesis we try to analyze the fusion of audio and video modalities from a new perspective by including more understanding about the relationships between those signals. Furthermore, we have applied techniques that are new in the joint processing of audio-visual signals, such as nonlinear diffusion based on partial differential equations (PDEs) and graph cut segmentation, which are more common in the image processing community.

1.4 Thesis Organization

As explained before, this thesis presents two novel audio-visual fusion methods based on very different fusion strategies. Each method is applied to a challenging problem in the joint audio-visual processing domain. The four main chapters of this thesis contain the two fusion methods and the two corresponding applications.

Chapter 2 introduces the first audio-visual fusion method, which is based on the sparse decomposition of audio and video signals over redundant dictionaries. A small number of functions (atoms) capture the main structures in each modality. In the audio case each atom represents a concentration of acoustic energy in the time-frequency plane, while in the video case it corresponds to a salient image structure and its temporal evolution (changes in location, scale and rotation). Then, the fusion step consists on assessing the co-occurrence of relevant events in audio and video modalities, which are defined respectively as the presence of a sound and the movement of a salient image structure. The video representation used in this chapter is the same than in [53]. However, our fusion method can be applied to more general problems in this domain due to the audio features that we use. In [53] the authors assume that the motion of an image structure is associated to the entire soundtrack in a time period, while in our case different image structures can be associated to different sounds in the same time interval. Thus, our fusion approach is still valid when simultaneous audio-visual sources are present.

In **Chapter 3** we propose a novel method to detect and separate the audio-visual sources that are present in a scene. In a first stage, the synchrony between relevant events in audio and video modalities is quantified using the fusion method presented in the previous chapter. According to

this co-occurrence measure, audio-visual sources are counted and located in the image using a robust clustering algorithm that groups video structures exhibiting strong correlations with the audio signal. Next, periods where each source is active alone are determined by assessing the correlation between the audio atoms and the already labelled video atoms. *Spectral* Gaussian Mixture Models (GMMs) characterizing the sources acoustic behavior are learnt in time periods where only one source is active in order to separate the audio signal in mixed periods. The proposed approach is extensively tested on synthetic and natural sequences composed of speakers and music instruments. Results show that the proposed method is able to successfully detect, localize, separate and reconstruct the present audio-visual sources.

Chapter 4 presents the second audio-visual fusion method. This nonlinear diffusion approach is able to naturally focus on parts of a video sequence that are relevant for applications in joint audio-visual processing. The diffusion process is controlled by a diffusion coefficient based on an estimate of the synchrony between video motion and audio energy at each point of the video volume. As a result, the information in video regions whose motion is not coherent with the soundtrack is iteratively removed. We propose a discretization scheme based on finite differences that ensures the stability of the proposed approach. To evaluate our method's efficiency, we introduce a quantitative measure that compares the strength of the diffusion process *inside* and *outside* the audio-related video regions. This performance measure is latter used to discuss appropriate values for the main parameter in our approach. Finally, an intuitive and computationally inexpensive stopping criterion for the audio-visual diffusion process is proposed. Our method is tested in challenging situations involving strong distracting motion and sequence degradation. Results show that in all cases our approach favors video regions related to the soundtrack.

In **Chapter 5** the audio-visual diffusion approach presented in the previous chapter is applied to the automatic extraction of audio-visual objects, i.e. the video regions that are related to the soundtrack. In a first stage, the coherence between audio and video channels is assessed by taking into account the strength of the audio-visual diffusion at each point of the video volume: the audio-video coherence is high in regions whose intensity is well preserved through the diffusion process while its value is low in the rest of the volume. Next, pixels with a very high audio-visual coherence are automatically labelled as belonging to the audio-visual object. Then, a 3D graph cut segmentation approach, which includes an audio-visual term linking together pixels presenting a high coherence, is used to extract the audio-visual objects. Since real sequences are very big, this segmentation procedure is applied within groups of frames (GoF) sequentially. Thus, the information about the source characteristics (location, shape and color statistics) in a GoF is propagated forward and integrated in the processing of the next GoF. The proposed approach is demonstrated on sequences presenting complex challenges such as non-stationary sources, distracting moving objects, and multiple sources with different activity patterns.

Finally, in **Chapter 6** we discuss in a global way the achievements of this thesis. Furthermore, some possible future research directions are proposed.

Audio-Visual Fusion based on Sparse Redundant Representations

2

2.1 Motivation

As discussed before, most audio-visual fusion approaches do not take into account the inherent structures in audio and video signals. The features that they use are either pixel intensities [13, 33, 76] or temporal variations [19, 36, 58] for the video, and audio energy [33, 36, 53] or cepstrum coefficients [10, 19, 58, 75] for the audio. Then, these representations are combined in a fusion step, whose objective is to assess the synchrony between both modalities using canonical correlation analysis [36, 75] or joint audio-visual probabilities for the features [10, 13, 26, 32, 33, 58]. Basically existing methods are trying to assess the coherence between each pixel's behavior and the entire audio signal (or a fragment of it when the analysis is performed using time shifting windows). As a result, those approaches do not exploit the spatio-temporal coherence that characterizes video signals, i.e. typically an image structure is composed of several pixels of similar characteristics and small variations appear between consecutive frames. Furthermore, many of these audio-visual fusion methods do not even consider the possibility that in a time slot the soundtrack can be composed of two or more sources. In this case, some sounds, i.e. peaks in the audio energy function, would be coherent with some video structure, while some other sounds (maybe happening approximately at the same time and composing thus the same energy peak) would be correlated to another part of the image.

Two main considerations have to be taken into account when attempting the fusion of audio and video modalities. First, sounds are not related to a single pixel's movement but rather to a set of pixels depicting an image structure. Second, one peak in the audio energy function can be composed of sounds coming from different sources, and this fact can not be neglected when evaluating the audio-video synchrony. The objective of this chapter is thus to understand first the physics in the problem in order to choose audio and video features more efficiently.

Let us now discuss the main characteristics of the considered modalities. The video signal can be seen as a set of image regions (delimited by edges) that evolve through time. Similarly, the audio signal can be decomposed into a set of sounds with different frequencies. Thus, if we define

appropriate functions we will be able to decompose each of the modalities into a small set of elements that have a physical sense. The basic functions that we consider in this chapter are sounds, i.e. concentrations of acoustic energy in the spectrogram and image structures (edges) evolving through time. In this chapter, the Matching Pursuit (MP) algorithm proposed by Mallat and Zhang in [51] is used for the decomposition of audio and video modalities. MP selects iteratively the elements of the dictionary (basic functions called atoms) that better approximate the original signal. The accuracy in the representation can be simply determined by choosing an appropriate number of atoms. Furthermore, the iterative characteristic of the MP algorithm makes it applicable to high dimensional signals.

The audio-visual fusion method that we present in this chapter is based on the decomposition of audio and video signals into a small set of basic structures. The video signal is decomposed into image structures and their temporal transformations using the 3D-MP algorithm proposed by Divorra and Vandergheynst in [23]. Since generating a redundant dictionary containing all possible image structures with all possible transformations would be impossible in practice, this approach selects the most representative image structures (edges) in the first frame and then tracks them from frame to frame. The audio signal is decomposed using the classic MP algorithm [51] into groups of energy distributed in the spectrogram, i.e. sounds with different frequencies happening at different times. Thus, audio and video representations have now a physical meaning. Then, in the fusion step the correlation between both modalities is determined at the atom level by assessing the degree of synchrony between the presence of a sound (audio feature) and an oscillatory movement of a relevant image structure (video feature). Since now audio and video signals are described in a compact way, the dimensionality of the problem is highly reduced and the fusion step is intuitive and computationally inexpensive. The resulting value, the audio-visual correlation score, quantifies thus the coherence between the features corresponding to each pair audio-video atom. A high value indicates a high probability that the sound and the movement of the video structure are related, since both events occur approximately at the same time.

The combination of the proposed video decomposition and features with only one audio feature (the audio energy evolution through time) has already been demonstrated to be appropriate for the joint processing of audio-visual signals in the sound source localization task by Monaci et al. in [53]. However, the computation of one-to-one relationships between audio and video basic functions (atoms) allows its applicability to more general problems in this domain. Thus, our approach can also deal with sequences where several sources are mixed: in our case the audio signal in a time period is not assumed to belong to only one speaker. This aspect represents a significant improvement over other methods in this domain.

This chapter is structured as follows. Section 2.2 introduces the main concepts for the sparse decomposition of signals over redundant dictionaries of functions. In Section 2.3 we detail the representations that are used for audio and video signals, both of them based on the Matching Pursuit algorithm. Next, Section 2.4 describes the proposed fusion method, which combines audio and video signals at the atom level. First, we introduce the features that capture relevant events associated to the atoms and then the degree of synchrony between audio and video events is quantified. In Section 2.5 we illustrate the applicability of the proposed representations to the joint processing of audio and video signals. For this purpose, we provide some examples of its application to sound source localization when using the proposed video representation and features combined with one audio feature, i.e. the estimated energy in the audio channel. The main possibilities of this approach are finally discussed in Section 2.6.

2.2 Sparse Representations over Redundant Dictionaries

Sparse representations are able to capture the most salient and meaningful structures that compose a signal by approximating it with a linear combination of a small number of elementary functions called atoms. Usually, these atoms are chosen from a dictionary composed of a *redundant* set of basis functions, whose objective is to capture the large variety of structures present in natural signals. Sparse representations have been successfully used in applications such as compression, feature extraction, noise reduction, regularization in inverse problems, blind source separation and pattern classification.

Let $\mathcal{D} = \{\phi_n\}_{n\in\Omega}$ be a *dictionary* of vectors called *atoms*, with $\phi_n : \mathbb{R}^d \to \mathbb{R}$ and $\|\phi_n\| = 1$ and Ω indexes the (finite) set of all functions composing $\mathcal{D}$. The idea is to approximate a digital signal $f \in \mathbb{R}^d$ by means of a linear combination of atoms belonging to the dictionary $\mathcal{D}$ as

$$f \approx \sum_{n\in\Gamma} c_n \phi_n \,, \tag{2.1}$$

where c_n is a coefficient weighting each component and Γ denotes a subset of atoms from the dictionary: $\Gamma \subset \Omega$.

A dictionary is said to be redundant when the number of atoms exceeds the dimension d of the signal space, so that any signal f can be represented by more than one combination of different atoms. Well-chosen redundant dictionaries have the capacity of providing representations that are *sparse*: a small number of atoms are required to approximate the signal f and thus the cardinality of Γ is much smaller than the dimension of the signal. Notice that the elements of the dictionary have to be very diverse in order to adapt better to the characteristics of the signal that we want to approximate.

As explained before, several combinations of atoms can be used in the representation of the signal f when using a redundant dictionary. In fact, the choice of the sparsest representation has been demonstrated to be an NP-hard problem [21]. As a result, a considerable effort has been put into the development of many sub-optimal schemes which consider approximate solutions. The pursuit algorithms that have been proposed can be divided into two main groups. The first group is composed by greedy algorithms that select the dictionary atoms sequentially, e.g. *Matching Pursuit* (MP) [51] and *Orthogonal Matching Pursuit* (OMP) [21, 61]. These methods are very simple since they involve basically inner products between the signal and the atoms in the dictionary. The algorithms composing the second group process all the coefficients simultaneously, e.g. *Basis Pursuit* (BP) [16] and the *Focal Underdetermined System Solver* (FOCUSS) family of algorithms [31]. These methods are more complex and computationally expensive. As a result, they are not suited for the analysis of high-dimensional signals such as the video sequences.

In this chapter we use the Matching Pursuit algorithm for the decomposition of audio and video signals. MP is an iterative greedy algorithm that selects at each step the dictionary element best correlated with the signal that we want to approximate. The major advantage of MP is that it admits simple and fast implementations, which are more suited for high-dimensional data. In this case, the signal approximation is built up by picking one coefficient at a time, while in BP and FOCUSS all the coefficients need to be chosen at the same time.

Let us now describe the Matching Pursuit algorithm. For further details, please refer to [51].

As explained before, Matching Pursuit decomposes any signal f into a linear combination of basis functions ϕ_n that are chosen from a redundant dictionary $\mathcal{D}$. For this purpose, MP iteratively selects the function in the dictionary that better approximates the signal. The first step of the

algorithm decomposes f as

$$f = R^0 f = \langle f, \phi_0 \rangle \phi_0 + R^1 f \,, \tag{2.2}$$

where $R^1 f$ is the residual component after projecting f in the subspace described by ϕ_0. All the elements in $\mathcal{D}$ have by definition a unit norm, and thus it is easy to see from equation (2.2) that ϕ_0 is orthogonal to $R^1 f$. This leads to

$$\|f\|^2 = |\langle f, \phi_0 \rangle|^2 + \|R^1 f\|^2 \,. \tag{2.3}$$

To minimize $\|R^1 f\|$ we must choose ϕ_0 such that the coefficient $|\langle f, \phi_0 \rangle|$ is maximum. At the next step, we select the element in the dictionary that better approximates the residual $\|R^1 f\|$ and thus we obtain

$$R^1 f = \langle R^1 f, \phi_1 \rangle \phi_1 + R^2 f \,. \tag{2.4}$$

The same procedure is applied recursively, and after N iterations f is decomposed as

$$f = \sum_{n=0}^{N-1} \langle R^n f, \phi_n \rangle \phi_n + R^N f \,. \tag{2.5}$$

Similarly, the energy $\|f\|^2$ is decomposed into

$$\|f\|^2 = \sum_{n=0}^{N-1} |\langle R^n f, \phi_n \rangle|^2 + \|R^N f\|^2 \,. \tag{2.6}$$

Thus, the original function f is decomposed into a linear combination of elements from the dictionary, which are chosen to best match the residual after each step of the approximation.

The MP algorithm has been shown to converge, i.e. $\|R^N f\|^2 \to 0$ when $N \to \infty$, and its approximation error decay rate has been shown to be bounded by an exponential [21]. As a result, the MP technique can be used to approximate the function f using N terms as

$$f \approx \sum_{n=0}^{N-1} c_n \phi_n \,, \tag{2.7}$$

where the coefficient $c_n = \langle R^n f, \phi_n \rangle$.

The iterative characteristic of the MP algorithm allows its application to the decomposition of high-dimensional signals. Since the basis functions (atoms) that are chosen from the dictionary are ranked according to their contribution to the approximation of the signal, this algorithm is *scalable*. Thus, the degree of accuracy of the signal approximation can be easily controlled by choosing the appropriate number N of functions from the dictionary.

2.3 Representations for Audio and Video Signals

Next subsections describe decomposition techniques used to represent in a compact way audio and video signals. Both modalities are iteratively decomposed by means of the Matching Pursuit algorithm that was described in last section. According to this signal decomposition, audio atoms represent concentrations of acoustic energy in the time-frequency plane (sounds), while video atoms represent image structures and the temporal transformations that they experiment. The proposed representations decompose thus the signals according to their salient structures, whose variations in characteristics such as dimensions or position represent a relevant change in the entire signal.

2.3.1 Audio Representation

The Matching Pursuit (MP) algorithm [51] is used to represent concisely the distribution of audio energy in the time-frequency plane. The audio signal $a(t)$ is decomposed over a dictionary of Gabor atoms $\mathcal{D}^{(a)}$, where a single window function, $g^{(a)}$, generates all the atoms that compose the dictionary. Each atom $\phi_k^{(a)} = U_k g^{(a)}$, is built by applying a transformation U_k to the mother function $g^{(a)}$, which is a normalized Gaussian window in our case. The possible transformations are scaling by $s > 0$, translation in time by u and modulation in frequency by ξ. Then, indicating with an index k the set of transformations (s, u, ξ), an audio atom can be represented as

$$\phi_k^{(a)}(t) = \frac{1}{\sqrt{s}}\, g^{(a)}\left(\frac{t-u}{s}\right) e^{i\xi t}\,, \tag{2.8}$$

where the value $1/\sqrt{s}$ makes $\phi_k^{(a)}(t)$ unitary. According to this definition, each audio atom represents a sound, i.e. a concentration of acoustic energy in the time-frequency plane around time u and frequency ξ.

Thus, an audio signal $a(t)$ can be approximated using K atoms as

$$a(t) \approx \sum_{k=0}^{K-1} c_k \phi_k^{(a)}(t)\,, \tag{2.9}$$

where c_k corresponds to the coefficient for every atom $\phi_k^{(a)}(t)$ from the dictionary $\mathcal{D}^{(a)}$.

Figure 2.1 shows an example of the decomposition of an audio signal into 3000 Gabor atoms. From top to bottom, we can observe the 1D audio signal, its energy distribution in the 2D time-frequency domain, and its MP decomposition into a relatively small set of atoms. The Matching Pursuit decomposition provides thus a sparse representation of the audio energy distribution in the time-frequency plane, highlighting the frequency components evolution. Moreover, MP performs a denoising of the input signal, pointing out the most relevant structures [51].

2.3.2 Video Representation

Ideally, we would like to represent the video modality using a small number of 3D functions capturing the salient geometrical structures in the signal. Let $v(x, y, t)$ be the video signal at pixel coordinates (x, y) and frame t. In this case the video signal decomposition would be expressed by

$$v(x, y, t) \approx \sum_{m \in \Gamma} c_m \phi_m^{(v)}(x, y, t)\,, \tag{2.10}$$

where c_m is the coefficient corresponding to each 3D video atom $\phi_m^{(v)}(x, y, t)$ and Γ denotes the subset of selected atoms from the dictionary $\mathcal{D}^{(v)}$. However, building such a dictionary is not feasible in practice. Notice that this dictionary should contain a huge number of 3D video structures since many possible temporal transformations need to be considered. In fact, even if we limit the set of transformations, an immense number of elements in the dictionary would still be required

As a result, another strategy needs to be considered for the video signal decomposition. For this purpose, we use the 3D-MP algorithm proposed by Divorra and Vandergheynst in [23]. This method decomposes the first video frame into a small set of 2D atoms and then tracks this atoms from frame to frame by allowing some transformations. Thus, in this case the video signal is decomposed into a set of atoms representing salient image components and their temporal transformations (i.e changes in their position, size and orientation). Unlike the case of simple pixel-based representations, when

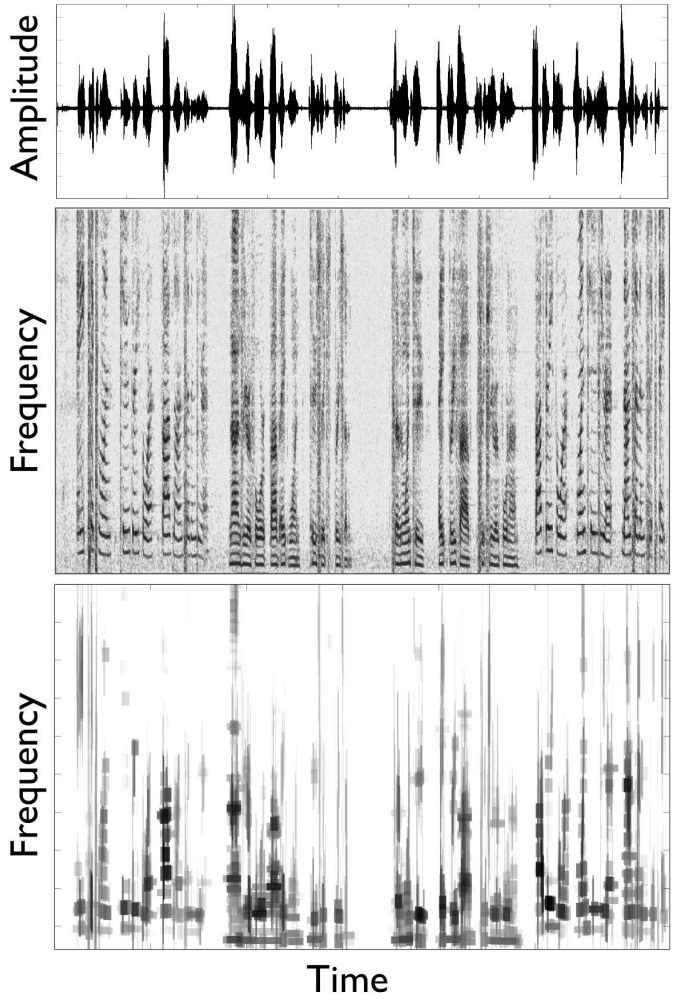


Figure 2.1 – *Audio signal decomposition into sparse redundant representations: Original audio signal [top], time-frequency energy distribution [middle] and corresponding Matching Pursuit approximation when using* 3000 *atoms of a Gabor dictionary [bottom].*

considering image structures that evolve smoothly through time we deal with dynamic features that have a true geometrical meaning. Furthermore, sparse geometric video decompositions provide compact representations of information, allowing a considerable dimensionality reduction of the input signals.

Let us now explain this video decomposition procedure. For further details, the interested reader can refer to [23].

In a first stage, the first frame of the video signal, $I_1(x, y)$, is approximated with a linear combination of atoms retrieved from a redundant dictionary $\mathcal{D}^{(v)}$ of 2D atoms as

$$I_1(x, y) \approx \sum_{m \in \Gamma} c_m \phi_m^{(v)}(x, y)\,, \tag{2.11}$$

where c_m is the coefficient corresponding to each 2D video atom $\phi_m^{(v)}(x, y)$ and Γ is the subset of selected atom indexes from the dictionary $\mathcal{D}^{(v)}$. As in the audio case, the dictionary is built by varying the parameters of a mother function. Here we use an edge-detector atom with odd symmetry,

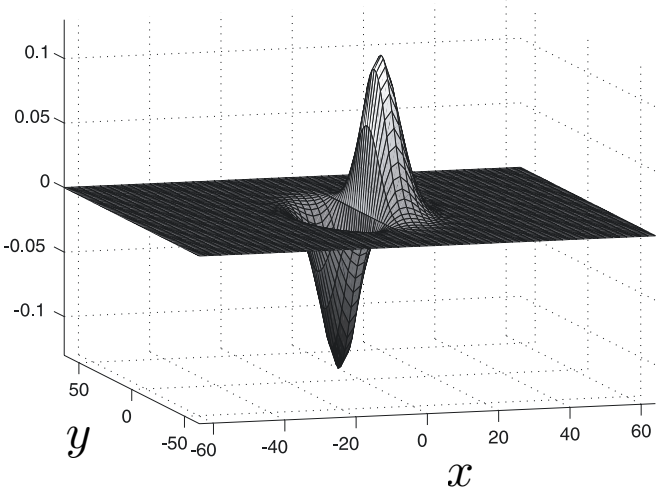


Figure 2.2 – *The video generating function* $g^{(v)}(x, y)$.

that is a Gaussian along one axis and the first derivative of a Gaussian along the perpendicular one (see Figure 2.2), so that it is able to approximate edges on the 2D image plane. The generating function $g^{(v)}$ is expressed as

$$g^{(v)}(x, y) = 2x \cdot e^{-(x^2+y^2)} \,. \tag{2.12}$$

As explained before, each 2D atom $\phi_m^{(v)} = U_m g^{(v)}$, is built by applying a transformation U_m to the mother function $g^{(v)}$. The possible transformations are translations over the image plane $\vec{r} = (r_1, r_2)$, scaling $\vec{s} = (s_1, s_2)$ to adapt the atom to the considered image structure and rotations θ to locally orient the function along the edge. Then, indicating with an index m the set of transformations $(r_1, r_2, s_1, s_2, \theta)$, a 2D atom can be represented as

$$\phi_m^{(v)}(x, y) = \frac{B}{\sqrt{s_1 s_2}} \cdot u_1 \cdot e^{-(u_1^2+u_2^2)} \,, \tag{2.13}$$

where the value B makes $\phi^{(v)}(x, y)$ unitary and

$$\begin{aligned} u_1 &= \frac{\cos\theta(x - r_1) + \sin\theta(y - r_2)}{s_1} \\ u_2 &= \frac{-\sin\theta(x - r_1) + \cos\theta(y - r_2)}{s_2} \,. \end{aligned}$$

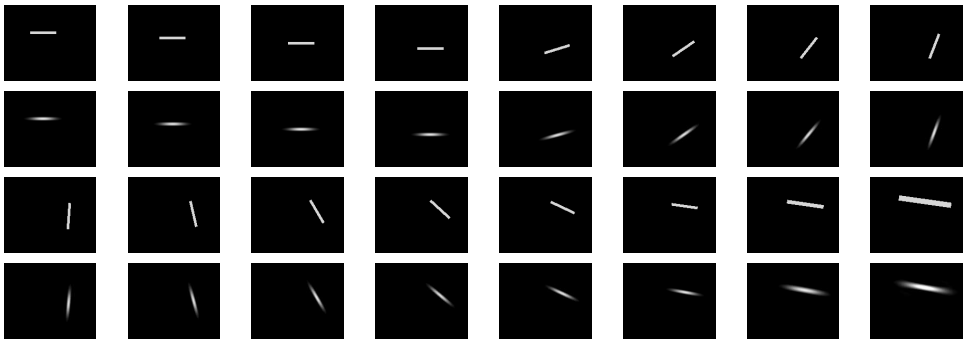
Then, each 2D atom is tracked from frame to frame using a modified MP approach based on a Bayesian decision criteria, which is explained in depth in [23].

As a result, the video signal can be approximated using M 3D video atoms $\phi_m^{(v)}$ as

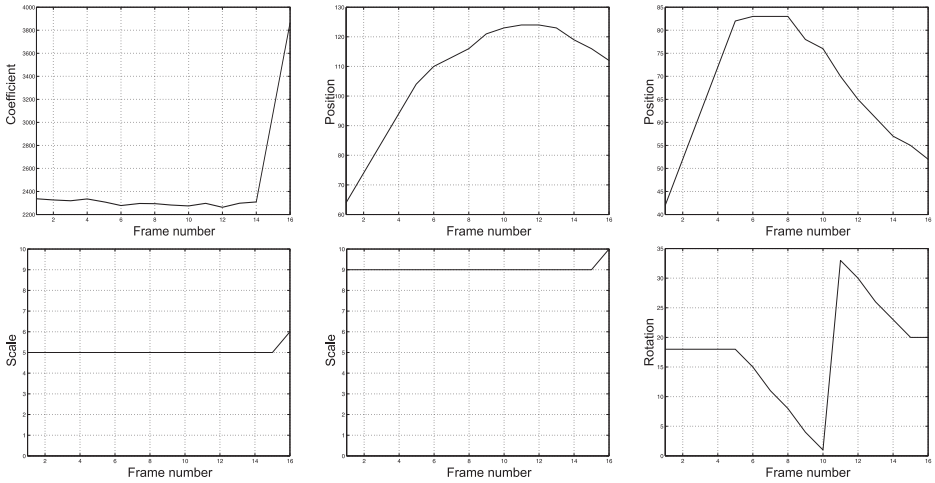
$$v(x, y, t) \approx \sum_{m=0}^{M-1} c_{m(t)} \phi_m^{(v)}(x, y, t) \,, \tag{2.14}$$

where the coefficients $c_{m(t)}$ vary through time and each video atom $\phi_m^{(v)}$ is obtained by changing from frame to frame the parameters $(r_{1_m}, r_{2_m}, s_{1_m}, s_{2_m}, \theta_m)$ of a reference 2D atom $\phi_m^{(v)}(x, y)$:

$$\phi_m^{(v)}(x, y, t) = \phi_{m(t)}^{(v)}(x, y) \,. \tag{2.15}$$



(a) Synthetic sequence approximated by 1 atom: first and third row show the original sequence made by a simple moving object. Second and fourth row depict the approximation using one video atom.



(b) Parameter evolution of the approximated object; from left to right and from up down, we find: coefficient c, horizontal position r_1, vertical position r_2, short axis scale s_1, long axis scale s_2, rotation θ.

Figure 2.3 – *Approximation of a synthetic scene by means of a 2D time-evolving atom*

An illustration of this video decomposition can be observed in Figure 2.3, where the approximation of a simple synthetic object by means of a single video atom is performed. Figure 2.3(a) shows the original sequence (top row) and its approximation composed of a single geometric term (bottom row). Figure 2.3(b) depicts the parametric representation of the sequence: we find the temporal evolution of the coefficient $c_{m(t)}$ and of the position, scale and orientation parameters. This 3D-MP video representation provides thus a parametrization of the signal which concisely represents the image geometric structures *and* their temporal evolution.

2.4 Audio-Video Atomic Fusion

In this section we describe the fusion method that allows us to quantify the relationships between audio and video modalities at the atom level. As explained before, approaches in joint audio-visual signal processing are based in an assumption of synchrony between related events in audio and video channels, e.g. when a person is speaking his/her lips movements are temporally correlated to the speech. In this work, we measure the degree of synchrony between audio and video features representing *relevant events* in each modality, which are defined respectively as the presence of

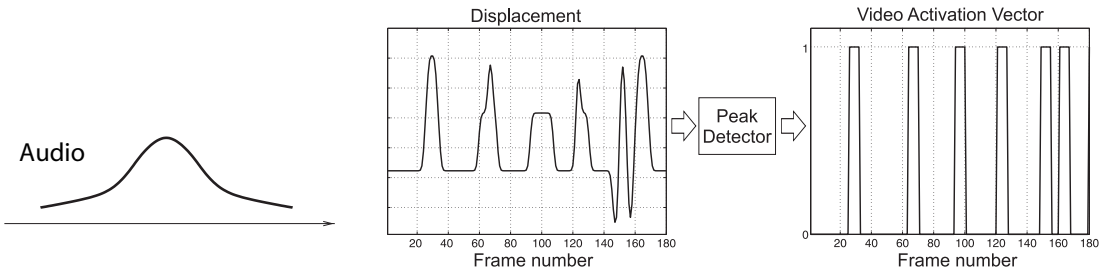


Figure 2.4 – *Audio feature $f_k^{(a)}(t)$ (a) and displacement function $d_m(t)$ with corresponding* Activation Vector *obtained for a video atom (b).*

an audio atom (energy in the time-frequency plane) and a peak in the video atom displacement (oscillation from an equilibrium position). Unlike previous methods in this domain, our high-level features are directly linked to the physics of the problem and, as a result, the synchrony between modalities can be assessed by means of a simple scalar product quantifying the co-occurrence of audio and video events.

For this purpose we first compute M video features corresponding to the M video atoms that represent the video modality and K audio features for the K audio atoms in which the audio modality is decomposed. As explained before, these features indicate the presence of a relevant event associated to the atom. Then *correlation scores* $\chi_{k,m}$ are computed between the features corresponding to each audio atom $\phi_k^{(a)}$ and each video atom $\phi_m^{(v)}$. The correlation scores quantify the synchrony between the events associated to audio atom k and video atom m, i.e. the correlation score is high if the considered image structure (video atom) is moving when that sound (audio atom) occurs.

In this section we explain first the features that we propose and then the fusion method that we use.

As explained before, audio and video features show the presence of a relevant event in each modality. Thus, those features depend only on the time index t, and their value is high when a relevant event takes place.

Audio feature:

A relevant event in the audio modality is the presence of a sound. As a result, we want an audio feature whose value is low all the time but it increases when the sound appears. In our case, each group of energy (sound) in the time-frequency plane is described by one atom. Thus, we only need to project this energy over the time axis.

The feature $\mathsf{f}_k^{(a)}(t)$ corresponding to audio atom k that we consider is the energy distribution of this atom projected over the time axis. In the case of Gabor atoms it is a Gaussian function whose position and variance depend on the atoms parameters u and s respectively. A scheme of this feature is shown in Figure 2.4(a). Notice that the audio feature does not depend on the frequency ξ in which the atom is centered.

Video feature:

An oscillatory movement of an image structure represents a relevant event in the video case. Thus, the value of the video feature should be low when the structure is static and high when it is moving. Since the video signal is already decomposed into a small set of image structures

evolving through time and we know the position of these structures at each moment, all we need to do is to compute their movement. In our case the video feature corresponding to each video atom will be defined by the peaks in the displacement of the atom.

Thus, for each video atom $\phi_m^{(v)}$ we first compute a feature describing its displacement $d_m(t) = \sqrt{r_{1_m}^2(t) + r_{2_m}^2(t)}$ by using the position parameters $(r_{1_m}(t), r_{2_m}(t))$ extracted from the tracking step of the decomposition at each frame t. Then, an *Activation Vector* [53] is built for each atom displacement function $d_m(t)$ by detecting the peaks locations as shown in Figure 2.4(b). The Activation Vector peaks that compose the video feature $\mathsf{f}_m^{(v)}(t)$ of each atom are filtered by a window of width $W = 13$ samples in order to model delays and uncertainty. Here $W = 13$ samples corresponds to 0.45 seconds, a time delay between audio and video *relevant events*, i.e. a movement and the presence of the corresponding sound, that appears to be appropriate.

It is important to clarify that the peaks of the displacement function $d_m(t)$ represent an oscillatory movement of the atom m. Thus, the video feature $\mathsf{f}_m^{(v)}(t)$ does not depend on the original or relative position of the video atom m in the image. Notice that the peaks are situated at the time instant where a change in the direction of the movement appears. That can be interpreted as a change in the sign of the acceleration of the atom or, what is the same, an oscillation on the movement of that atom.

At this point the features corresponding to each audio and video atom are already computed. An intuitive way to combine them and quantify the synchrony between relevant events is simply to compute the scalar product between features. Thus, the resulting value represents a measure of the temporal overlap between the features corresponding to audio and video atoms or, in other terms, the synchrony between the motion of a video structure and a sound with a certain temporal tolerance modeled by W.

The *correlation score* $\chi_{k,m}$ between audio atom k and video atom m is defined as

$$\chi_{k,m} = \langle \mathsf{f}_k^{(a)}(t), \mathsf{f}_m^{(v)}(t) \rangle \ . \tag{2.16}$$

This value is high when the audio atom and a peak in the video atom's displacement overlap in time, i.e. when a sound (audio energy) occurs more or less at the same time than the video structure is moving. Thus, a high correlation score means high probability for a video structure of having generated the sound.

2.5 Evaluation

The video signal decomposition in terms of redundant representations has already been demonstrated to provide good results in the audio-visual speaker localization task in [53]. The video feature that they use is exactly the same that we have presented in this section. In contrast, they use simply an estimation of the energy in the audio channel as a feature for the audio signal. The fusion step in their case measures thus the synchrony between each video atom's displacement peaks and the audio energy. Then, they localize the current speaker by detecting the video atom more correlated to the soundtrack. Figure 2.5 shows some results obtained in the speaker localization task in [53]. We include them in this thesis for completeness. The analyzed clips depict either one person standing in front of a camera and reading digit strings in (a), (b), or two persons, only one of which is speaking in (c), (d). The results show that the methodology they propose allows to locate and track the speaker's mouth in sequences where the speaker is moving (Figures 2.5(a), (b)) and sequences presenting distracting motion generated by another person (Figures 2.5(c), (d)). As expected, in

(a) (b) (c) (d)

Figure 2.5 – *Results obtained when applying the method in [53] to the speaker localization task. The first row shows the original video frames and the second row shows the white footprints of the video atoms that present the highest correlation with the corresponding audio signal. In all cases, the speaker's mouth is correctly detected.*

all cases the algorithm chooses video atoms that constitute the mouth and/or chin structures of the current speaker.

Thus, the suitability of the video signal decomposition in terms of sparse redundant representations to joint audio-visual processing has already been demonstrated. In contrast, the audio representation that we proposed in Section 2.3.1 has not been tested yet. Notice that our audio feature is more complex than the estimated audio energy in [53], since in this work we consider each sound separately. Indeed, each audio atom represents a concentration of energy in the time frequency plane. Thus, each atom contains information about when the sound occurs and which frequency it has. As a result, this audio atomic representation can deal with more complex problems. Specifically, we demonstrate in Chapter 3 its applicability in the audio-visual source separation task. In [53], the authors assume implicitly that there is only one sound source in a time slot and, as a result, their objective is to simply locate *the* sound source in the image. In contrast, our goal is to extract the audio and video part that compose each sound source in sequences where *several* sound sources are present, i.e. we want to localize and separate multiple simultaneous sources. The proposed audio and video decompositions into redundant dictionaries of atoms and the fusion step that evaluates the correlation between audio and video relevant events are thus demonstrated in Chapter 3 to be a useful tool to analyze audio-visual sequences.

2.6 Discussion

In this chapter we have presented an audio-visual fusion method based on the sparse decomposition of both modalities into redundant dictionaries of functions. Audio and video signals are thus represented as the sum of a small number of atoms having a physical meaning: an audio atom represents a sound and a video atom captures a moving image structure. Then, we define audio and video features according to the relevant events in each modality, which are respectively the oscillatory movement of an image structure and the presence of a sound. Finally, an audio-visual correlation measure is introduced to quantify the degree of synchrony between the features corresponding to each audio atom and each video atom. The higher is this measure the higher is the probability for

the moving video structure to having generated that sound, i.e. the higher is the probability that those events are caused by the same physical phenomenon.

The video representation and the corresponding features that we have introduced in this chapter have already been proven in [53] to provide good results in the audio-visual sound source localization task. However, in their case only one audio feature is extracted from the soundtrack, i.e. an estimation of the energy in the audio channel. As a result, they evaluate the coherence between each video atom's feature and the energy in the entire audio signal. Analogously to the video case, we have proposed an audio feature for each audio atom, leading thus to a one-to-one relationship when assessing the audio-video coherence. This configuration gives more flexibility to our approach. The most important advantage is that our fusion method is still valid when two (or more) sound sources contribute to the soundtrack. In this case, some video atoms would be related to the audio atoms (sounds) generated by one source while other video atoms would be linked to the sounds belonging to the other sound source. In the next chapter we apply this video fusion method to the audio-visual separation of a mixture of sources, i.e. we extract the audio and video parts corresponding to each source in the scene with no interference between them.

Application: Blind Audio-Visual Source Separation

3

3.1 Introduction

In Section 2.5 we have shown the applicability of the audio-visual modeling using sparse redundant representations to the sound source localization task. The considered sequences contained only one audio-visual source (speaker) and in some of them also a visual distractor (a silent person) was present. However, in an unconstrained environment audio-visual sequences can be much more complex. Typically, they can be composed by more than one audio-visual source, and maybe some audio-only sources (whose sounds are not linked to any moving object in the camera's field of view). For example, that would be the case when two persons are speaking and the sounds of a radio are present in the background. Notice that some visual distractors can be present too, i.e. moving objects without any associated sound, but in this case we do not call them video-only sources since they are not the source of any sound. In this chapter we consider sequences where two audio-visual sources are mixed. Our objective is to extract separately each of the sources, by separating their audio and video modalities. Thus, in the separated sequences only one source is present and the other one is not interfering anymore. For simplicity, audio-only sources are not treated specifically in this analysis. Their contribution to the soundtrack is considered as noise and their audio energy will be present in the separated soundtracks corresponding to the audio-visual sources in the scene. Let us consider the example of a meeting. The scene is composed of several people speaking in turns or, sometimes, having parallel conversations. Detecting the current speaker/speakers and associating to each one of them the correct audio portions is extremely useful. For example, one could select one person and obtain the corresponding speech and image without the interference of other speakers. It can then be possible to index the whole meeting by using a speech-to-text algorithm. In this way one can search through amounts of indexed data by key-words and recover the target scene (or the person or exact date where the word appeared for example). The core of all these applications is the audio-visual source separation. In this chapter we present a new algorithm which is able to

The work presented in this chapter has been published in [45]. Some preliminary works on this subject can also be found in [42–44].

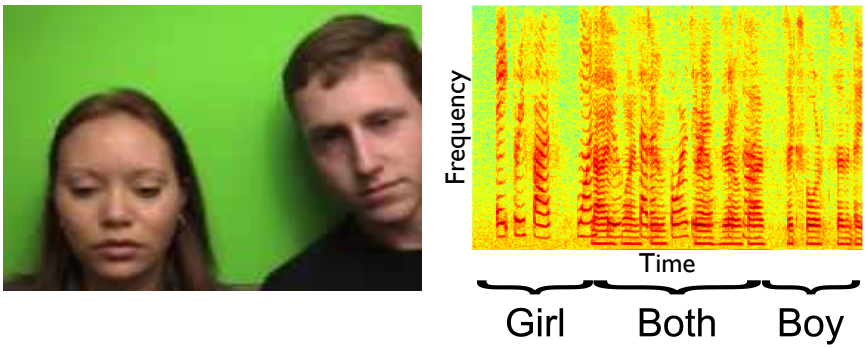


Figure 3.1 – *Example of a sequence considered in this work. The sample frame [left] shows the two speakers; as highlighted on the audio spectrogram [right], in the first part of the clip only the girl on the left speaks, then the boy on the right starts to speak as well, and finally the girl stops speaking and the boy speaks alone.*

automatically detect and separate the audio-visual sources that compose a scene.

One typical sequence that we consider in this work, taken from the *groups* section of the CUAVE database [62], is shown in Figure 3.1. It involves two speakers arranged as in Figure 3.1 [left] that utter digits in English. As highlighted in Figure 3.1 [right], in the first part of the clip only the girl on the left speaks, then the boy on the right starts to speak as well, and finally the girl stops speaking and the boy speaks alone. In this case, one audio-visual source is composed of the image of one speaker and the sounds that she/he produces. However, we must not associate to this source a part of the image (or soundtrack) belonging to the other speaker. What we want to do here is to detect and separate these audio-visual sources.

In a first stage towards a complete audio-visual source separation, several methods exploited synchrony between audio and video channels to improve the results in the *audio* source separation domain when *two* microphones are available [20, 66, 67, 77, 84]. In [66] the audio activity for each source (speaker) is assessed by computing the amount of motion in a previously detected mouth region. Then, the sources activity is used to improve the audio separation results when important noise is present. This method can only be used in speech mixtures recorded with more than one microphone. Approaches described in [20, 67, 77, 84] first build audio-visual models for each source and then they use them to separate a given *audio* mixture. For those last methods, the sources in the mixture and the video part of each one of them need to be known in advance, and the audio-visual source model is also built off-line.

Only two methods attempt a complete audio-visual source separation using a video signal and the corresponding *one-microphone* soundtrack [6, 73]. Barzelay and Schechner propose in [6] to assess the temporal correlation between audio and video *onsets*, which are respectively the beginning of a sound and a significant change on the speed or direction of a video structure. Audio-visual objects (AVO) are assumed to be composed of the video structures whose onsets match a majority of audio onsets and the audio signal associated to those audio onsets. The audio part corresponding to each AVO is computed by tracking the frequency formants that follow the presence of its audio onsets. In [73] a similar approach using canonical correlation analysis for finding correlated components in audio and video is presented. This approach uses trajectories of "interest" points in the same way as in [6] and it adds an implementation using microphone arrays. The main differences between those approaches and our method are the following:

1. The objective of the proposed method is to separate and reconstruct *audio-visual* sources. We

want to stress that our sources are audio-visual, not only audio or video. Existing methods do that only partially: they locate the video structures more correlated with the audio and separate the audio (in [73] there is no evidence however). Both methods do not attempt to reconstruct the video part of the sources. Concerning the audio, in [6] the separated soundtracks are recovered with an important energy loss due to the formants tracking, while in [73] no separated soundtracks are shown or analyzed.

2. We separate audio-visual sources using a simple and very important observation: it is very unlikely that sources are mixed all the time. Thus we detect periods during which audio-visual sources are active alone and periods during which they are mixed. This is a very important step because once one has this information, any one microphone *audio* source separation technique can be used. Thus, we do not need to know in advance the characteristics of the sources composing the mixture (off-line training is not needed anymore), since acoustic models for the sources can be learnt in periods where they are active alone.

In this research work, the robust separation of audio-visual sources is achieved by solving four consecutive tasks. First, we estimate the number of audio-visual sources present in the sequence (i.e. one silent person cannot be considered as a source). Second, the visual part of these sources is localized in the image. Third, we detect the temporal periods during which each audio-visual source is active alone. Finally, these time slots are used to build audio models for the sources and separate the original soundtrack when several sources are active at the same time. From a purely audio point of view, the video information ensures the blindness of the one microphone GMM-based audio source separation that will be explained in Section 3.4.2. The number of sources in the sequence and their characteristics are determined by combing audio and video signals. As a result, our algorithm does not need any previous information or off-line training to separate the audio mixture and accomplish the whole audio-visual source separation task.

This chapter has the following structure: in Section 3.2 we describe the *Blind Audio-Visual Source Separation* (BAVSS) algorithm, based on the redundant representations for audio and video signals that were previously introduced in Chapter 2. Next, Sections 3.3 and 3.4 explain in depth the methodology used for the separation of the video and audio parts of the sources respectively. In Section 3.5 we introduce the performance measures that are used in the evaluation of our method. Section 3.6 presents the separation results obtained on real and synthesized audio-visual clips. Finally, in Section 3.7 achievements and future research directions are discussed.

3.2 Blind Audio-Visual Source Separation Overview

Figure 3.2 schematically illustrates the whole BAVSS process. We observe N audio-visual sources, each one composed of its visual part and its audio part. Thus, the soundtrack contains the contribution of the N sources $a(t) = \{a_1(t), a_2(t), \ldots, a_N(t)\}$, and the video modality contains the video part of the N sources $v(x, y, t) = \{v_1(x, y, t), v_2(x, y, t), \ldots, v_N(x, y, t)\}$. Audio and video signals are decomposed using redundant representations into K audio *atoms* $\phi_k^{(a)}(t)$ and M video *atoms* $\phi_m^{(v)}(x, y, t)$ respectively, as explained in Section 2.3. Audio and video atoms describe meaningful structures in each modality in a compact way: an audio atom indicates the presence of a sound and each video atom represents a part of the image and its evolution through time.

In the next block, the fusion between audio and video modalities is performed at the atom level by assessing the temporal synchrony between the presence of a sound and an oscillatory movement of a video structure as explained in Section 2.4. The result is a set of correlation scores $\chi_{k,m}$ that

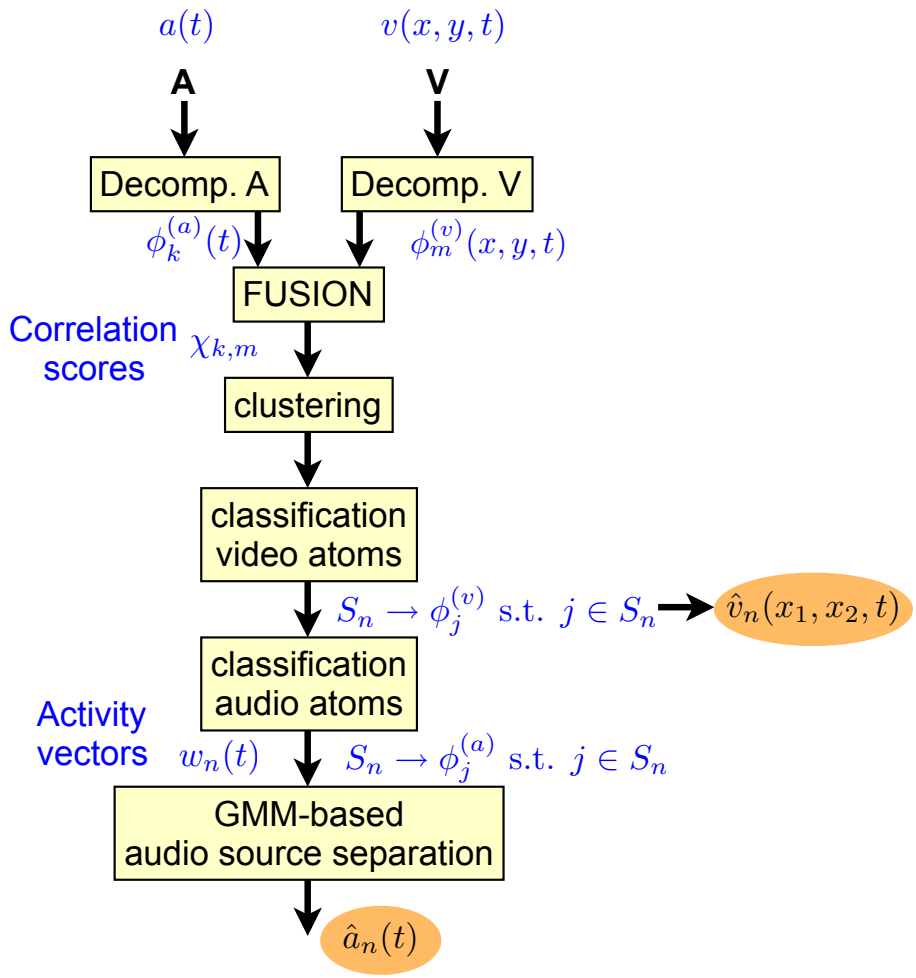


Figure 3.2 – *Block diagram of the proposed blind audio-visual source separation algorithm. Audio and video channels are* **decomposed** *using redundant representations. Temporal correlation between* relevant events *in both modalities is assessed and quantified in the* **fusion** *stage, giving as a result the correlation scores* $\chi_{k,m}$ *between audio and video atoms. Next, video atoms that present strong correlations with the soundtrack are grouped together using a* **clustering** *algorithm that determines the number of audio-visual sources* N *in the scene and locates them on the image. Then, video atoms are assigned to the corresponding sources using a proximity criterion, which provides an estimation of the video part of the sources. At this point, audio atoms are classified into the sources taking into account their correlation with the labelled video atoms. The activity of each source (represented by activity vectors in the diagram) is determined according to the audio atoms classification. Finally,* spectral *GMMs for the sources are built in temporal periods where the sources are active alone and these models are used to separate sources when they are mixed. In this way the audio part of the sources is also estimated and the process is completed.*

associate each audio atom k to each video atom m according to their synchrony.

Next, audio-visual sources are counted and localized using a clustering algorithm that spatially groups video structures whose movement is synchronous with the presence of sounds in the audio channel (Section 3.3.1). These initial steps are the most important ones for the BAVSS process since they assess the relationships between audio and video structures and determine the number N of present audio-visual sources. Thus, in order to recover an estimate of the video part of each

source we only need to assign the video atoms to the sources taking into account their positions in the image (the procedure is detailed in Section 3.3.2).

Then, each audio atom is assigned to one source according to the classification of the associated video atoms. However, this labelling of the audio atoms is not sufficient to clearly separate the audio sources. This is due to the fact that until this point our method only assesses the temporal synchrony between audio and video structures, and thus it is not discriminant when several sources are mixed. Thus we use the audio atoms classification to detect the temporal periods of activity of each source as explained in Section 3.4.1. The audio mixture is separated according to the *spectral* Gaussian Mixture Models that are built in time slots during which each source is active alone (Section 3.4.2). In this final step we obtain the estimates for the audio part of the sources and the complete audio-visual separation is achieved. The choice of the GMMs for the audio separation is motivated by their simplicity and the fact that GMMs can effectively represent the variety of sounds structures [8]. However, once the periods of activity of the sources are determined any one microphone audio source separation algorithm can be used.

An intuitive schema explaining the proposed BAVSS method is provided in Figure 3.3. First three rows illustrate the process used to locate and separate the video part of the sources in the image. First, the localization is performed by using a clustering algorithm that spatially groups the video structures in the image temporally correlated with the audio atoms of the soundtrack. Next, a purely spatial criterion is used to separate the sources. The last two rows show the audio source separation part. In a first stage, the correlations between audio and video events are employed to identify temporal periods with only one source active (audio source localization). Then, the sources frequency behavior is learnt in these periods and used to separate the sources in the mixed periods.

Two main assumptions are made on the type of sequences that we can analyze. First, as explained before we assume that for each detected video source there is one and only one associated source in the audio mixture. This means that if there is an audio-only source in the sequence (e.g. a person speaking out of the camera's field of view), it is considered as noise and its contribution to the soundtrack is associated to the sources found in the video. This assumption simplifies the analysis, since we know in advance that a one-to-one relationship between audio and video entities exists. The relaxation of this assumption will be the object of future investigation. Moreover, we consider the video sources approximately static globally, i.e. their location over the image plane do not change too much (sources never switch their positions for example). Again, this second assumption is made for simplicity and it can be removed by using a 3D clustering of the video atoms (using also the temporal dimension) instead of a 2D clustering. The video decomposition gives the position of the atom at each time instant and thus we can group together atoms that stay close through time to the video atoms most correlated to the soundtrack.

3.3 Video Separation

This section aims to provide a compact visual representation of each audio-visual source in the sequence by assigning to the source the video atoms that compose it. Thus, we first need to quantify the number of audio-visual sources in the scene, next we can classify the video atoms into the corresponding source and finally the video modality of each source will be reconstructed. Subsection 3.3.1 describes the procedure that counts the audio-visual sources and locates them in the image plane. It is based on the observation that the video modality of each source is composed of groups of atoms which are close to each other and present a high correlation with the audio signal. Thus, this step is achieved by using a clustering algorithm that takes into account the correlation

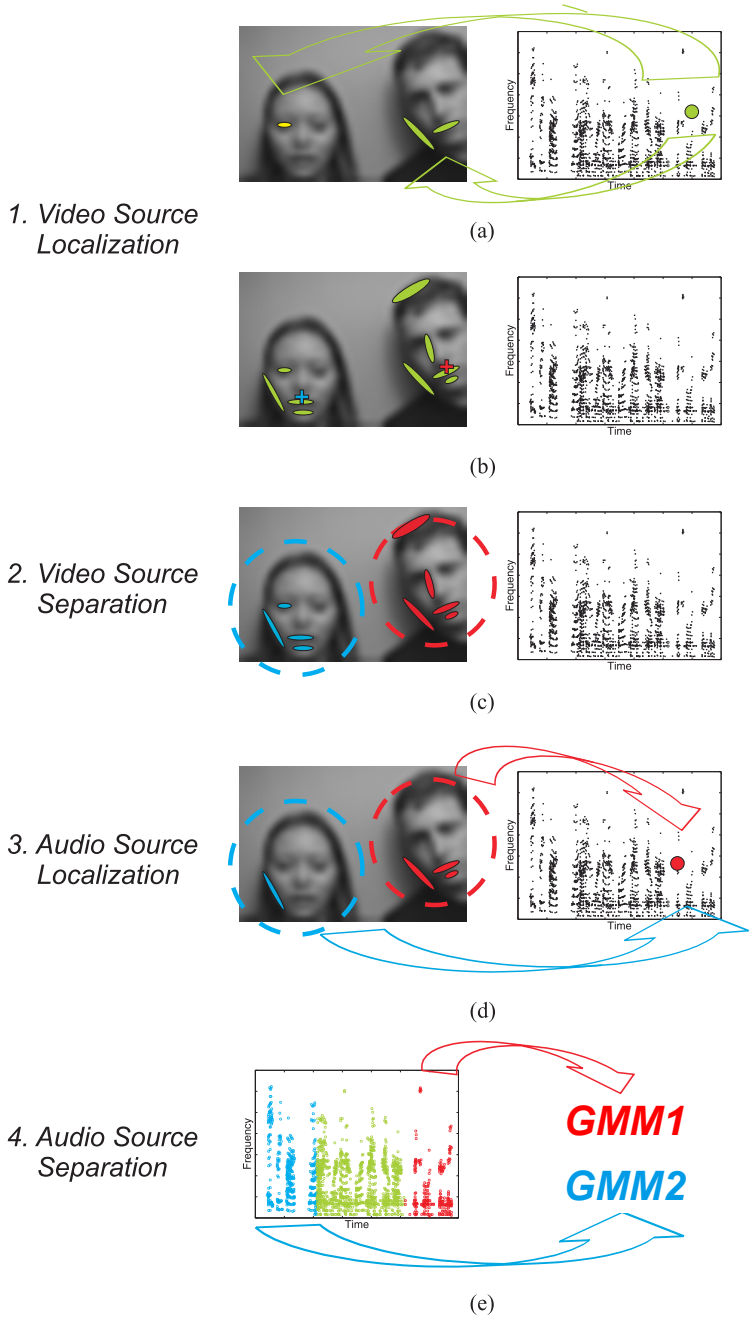


Figure 3.3 – *Schema of the audio-visual source separation algorithm.* Phase 1*: in (a) audio atoms (green dot on the spectrogram) are correlated with video atoms (green and yellow footprints are highlighted on the left image) and exploiting this information on picture (b) video sources are localized (blue and red crosses).* Phase 2*: video atoms are classified into the corresponding sources (c), as highlighted by the footprints colors (blue for the left speaker and red for the right one).* Phase 3*: audio atoms (red dot on the right) are classified into the corresponding sources using the audio-visual association information (d). Periods with only one active audio-visual source are detected.* Phase 4*: in temporal periods when a single source is active (blue and red markers) GMMs for each source frequency characteristics are estimated (e). These models are used to separate the sources in mixed periods (green markers).*

(synchrony) between audio and video atoms, which is obtained using the method described in Chapter 2. Once we have the number and location of the sources, the video atoms are labelled using a proximity criterion and the video modality of each source is reconstructed by adding the contribution of the corresponding video atoms (see Subsection 3.3.2).

3.3.1 Spatial Clustering of Video Atoms

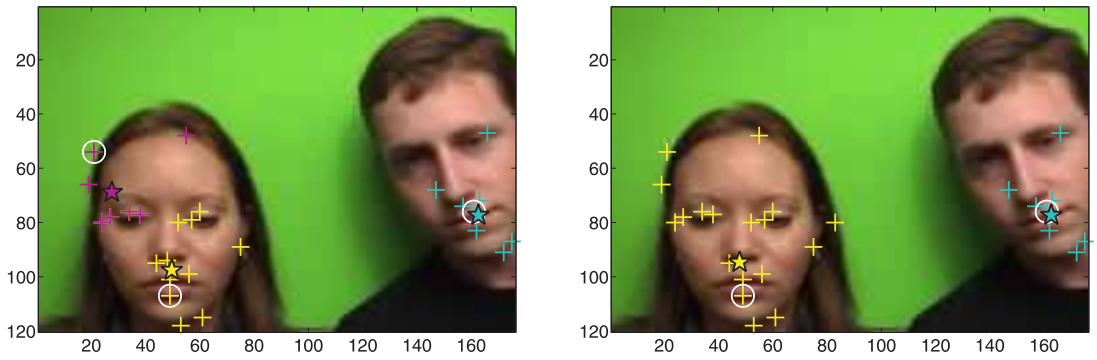
The idea now is to spatially group all the structures belonging to the same source in order to estimate the source position on the image. We define the empirical *confidence value* κ_m of the m-th video atom as the sum of the MP coefficients c_k of all the audio atoms associated to it in the whole sequence, $\kappa_m = \sum_k c_k$, with k such that the correlation score $\chi_{k,m} \neq 0$. This value is a measure of the number of audio atoms related to this video structure and their weight in the MP decomposition of the audio track. Thus, a video atom m whose motion presents a high synchrony with sounds in the audio channel will have a high confidence value κ_m, since a large number of important audio atoms in the sequence will be associated to this video atom in the audio-video atomic fusion step (Section 2.4). In contrast, low values for k_m correspond to video atoms whose motion is occasionally (and not continually) synchronous to the sounds.

Typically, the video part of each source is composed of groups of atoms presenting high confidence values κ_m (and thus high coherence with the audio signal), which are concentrated in a small region in the image plane. Thus, a spatial clustering becomes a natural way to count the sources and estimate their position in the image. Let each video atom m be characterized by its position over the image plane and its confidence value, i.e. $((r_{1_m}, r_{2_m}), \kappa_m)$. In this work, we cluster the video atoms correlated with the audio signal (i.e. with $\kappa_m \neq 0$) following these three steps:

1. **Clusters Creation:** The algorithm creates Z clusters $\{C_i\}_{i=1}^{Z}$, by iteratively selecting the video atoms with highest confidence value (and thus highest coherence with the audio track) and adding to them video atoms closer than a *cluster size* R defined in pixels. Video atoms belonging to a cluster can not be the center of a new cluster. Thus each new cluster is generated by the video atom with highest confidence value from those which have not been classified yet. Let $P = \{((r_{1_m}, r_{2_m}), \kappa_m)\}_m$ be the set of points (atoms) to be classified. The clusters are thus created using the following algorithm:

    1. Initialization : $Z = 0$, $P_Z = P_0 = P$;
    2. Find the point $((\tilde{r}_{1_m}, \tilde{r}_{2_m}), \tilde{\kappa}_m) \in P_Z$ with highest confidence value. It has the most important audio atoms associated, and consequently this video atom is the most probable to be the center of a source;
    3. Create a new cluster C_Z aggregating all the video atoms that are closer than a spatial maximum distance to $(\tilde{r}_{1_m}, \tilde{r}_{2_m})$ (*cluster size* R defined in pixels);
    4. Remove all the video atoms assigned to this cluster from the set of points to be classified, i.e. $P_{Z+1} = P_Z \setminus C_Z$;
    5. Stop the algorithm if all the points with confidence over the mean are already classified, otherwise increment $Z \leftarrow Z + 1$ and go back to step 2. Only video atoms with significant confidence value (highly correlated with the audio) can be the center of a new cluster.

2. **Centroids Estimation:** The center of mass of each cluster is computed taking the confidence value of every atom as the mass. The resulting centroids are the coordinates in the image where the algorithm locates the audio-visual sources;



(a) Clusters creation when $R = 40$ pixels

(b) Clusters creation when $R = 60$ pixels

Figure 3.4 – *Clusters created using different cluster sizes. The atom represented with a white circle (∘) is the one with higher* confidence value *that builds the cluster. Crosses (+) represent the coordinates of the video atoms aggregated to the cluster. Finally, the centroids of each cluster are indicated by a star (⋆). Each cluster is represented with a different color, from the first to last created (descendent importance of the cluster): yellow, cyan and the last one, magenta, which is present only on picture (a). Actually, the magenta cluster will be classified as* unreliable *and eliminated at the next step of the processing.*

3. ***Unreliable Clusters* Elimination:** We define the *cluster confidence value* K_{C_i} as the sum of the confidence values κ_j of the atoms belonging to the cluster C_i, i.e. $K_{C_i} = \sum_{j \in C_i} \kappa_j$. Based on this measure, *unreliable clusters*, i.e. clusters with small confidence value K_{C_i} are removed, obtaining the final set of $N \leq Z$ clusters, $\{C'_n\}_{n=1}^{N}$, with centroids (x_n, y_n). In this step we remove cluster C_i if

$$K_{C_i} < 0.1 \cdot \max_h K_{C_h} \quad \text{with } h = 1, \ldots, Z\,,\ h \neq i\,. \tag{3.1}$$

Figure 3.4 shows an example of applying this clustering procedure to a sequence with two speakers for cluster sizes $R = 40$ and $R = 60$. This figure illustrates the necessity of eliminating clusters with small confidence value. In general, when we decrease the *cluster size* R more possible sources appear (Z increases), but all these clusters are far from the mouth and present a small correlation with the audio signal (e.g. the *magenta* cluster in Fig. 3.4(a)). Thus, step 3 of the algorithm easily removes clusters that do not represent an audio-visual source since their confidence K_{C_i} is much smaller. In fact, the results are not significantly affected by the cluster parameters choice. For R ranging between 40 and 90 pixels the proposed clustering algorithm has been proved to detect the correct number of sources N (in all experiments image dimensions are 120×176 pixels). Further details about this clustering algorithm can be found in [42].

At this stage a good localization of the audio-visual sources in the image is achieved. The number of sources N does not have to be specified in advance since a confidence measure is introduced to automatically eliminate unreliable clusters.

3.3.2 Video Atoms Classification and Source Reconstruction

This step classifies *all* video atoms closer than the cluster size R to a centroid into the corresponding source. Notice that only video atoms moving coherently with sounds ($\kappa_m \neq 0$) are considered for the video localization in Section 3.3.1. Each such group of video atoms describes the video modality of an audio-visual source, achieving thus the video separation objective. Then, an estimate of the

Figure 3.5 – *Example of the video sources reconstruction. On the left picture the left person is speaking while on the right picture the right person is speaking.*

video part of the $n-$th source, S_n, can be computed simply as

$$\hat{v}_n(x,y,t) = \sum_{j \in S_n} c_{j(t)} \phi_j^{(v)}(x,y,t) \ . \tag{3.2}$$

Figure 3.5 shows an example of the reconstruction of the current speaker detected by the algorithm. Only video atoms close to the sources estimated by the presented technique are considered. Thus, to carry out the reconstruction, the algorithm adds their energy and the effect is a highlight of the speaker's face. In both frames, the correct speaker is detected.

3.4 Audio Separation

In this section we attempt the complete separation of the audio modality of the sources. Thus, for each audio-visual source in the scene we want to extract a soundtrack containing the sounds generated by this source without any *interference* coming from other sources. A first step described in Subsection 3.4.1 classifies the audio atoms into the corresponding source by evaluating the label of the video atoms that they are linked with (we again use the correlation between audio and video atoms from Chapter 2). Once this classification is achieved we can easily determine when only one source is active by localizing the time slots where this source owns a great majority of the audio atoms. Since the separation of the audio modality of the sources is extremely challenging in time slots where several sources are mixed, the frequency characteristics of the sources need to be determined. Subsection 3.4.2 details the learning process applied in time slots where sources are isolated (only one source is active), which provides an acoustic model for each source that can be used to separate their contributions in mixed periods.

3.4.1 Audio Atoms Classification

For every audio atom we take into account all related video atoms, their correlation scores and their classification into a source. Accordingly, an audio atom should be assigned to the source gathering most video atoms. Since we also want to reward synchrony, the assignation of each audio entity $\phi_k^{(a)}$ is performed in the following way:

1. Take all the video atoms $\phi_m^{(v)}$ correlated with the audio atom $\phi_k^{(a)}$, i.e. for which $\chi_{k,m} \neq 0$;

2. Each of these video atoms is associated to an audio-visual source S_n ; for each source S_n compute a value H_{S_n} that is the sum of the correlation scores between the audio atom $\phi_k^{(a)}$ and the

video atoms $\phi_j^{(v)}$ s.t. $j \in S_n$:

$$H_{S_n} = \sum_{j \in S_n} \chi_{k,j} \,; \tag{3.3}$$

Thus, this step rewards sources whose video atoms present a high synchrony with the considered audio atom.

3. Classify the audio atom into the source S_n if the value H_{S_n} is "big enough": here we require H_{S_n} to be twice as big as any other value H_{S_h} for the other sources. Thus we attribute $\phi_k^{(a)}$ to S_n if

$$H_{S_n} > 2 \cdot H_{S_h} \quad \text{with } h = 1, \ldots, N \,,\; h \neq n \,. \tag{3.4}$$

If this condition is not fulfilled (this is typically the case when several sources are simultaneously active), this audio atom can belong to several sources and further processing is required. This decision bound is not a very critical parameter since it only affects the classification of the audio atoms in time slots with several active sources. In periods with only one source, the difference between the score for the considered source H_{S_n} and the others is enormous and it is thus easy to classify the atom into the correct source.

Using the labels of audio atoms, time periods during which only one source is active are clearly determined. This is done using a very simple criterion: if in a continuous time slot longer than Δ seconds all audio atoms are assigned to source S_n, then during this period only source S_n is active. In all experiments the value of Δ is set to 1 second. The choice of this parameter has been done according to the length of the analyzed sequences (around 20 seconds). This value has to be small enough to ensure that in a period there is *only one* source active. At the same time, it has to be big enough to allow the presence of periods where to train the source audio models. Thus, Δ could be set automatically according to the length of the analyzed clip, e.g. one tenth of the sequence length.

When several sources are present, temporal information alone is not sufficient to discriminate different audio sources in the mixture. To overcome this limitation, in these *ambiguous* time slots a time-frequency analysis is performed, which is presented in details in the next section.

3.4.2 GMM-based Audio Source Separation

As explained in Section 3.2, the choice of the *spectral* Gaussian Mixture Models (GMMs) as our method for the separation of the audio part of the sources has been motivated by two main reasons. Inspite of its simplicity, we can achieve a good audio separation since GMMs are able to model multiple Power Spectral Densities (PSD) or, what is the same, several frequency behaviors for the same source. This is a very interesting property given the diverse nature of sounds. Thus GMMs have the capacity of modelling non-stationary signals contrary to classical Wiener filters [8].

Here, we perform a one microphone GMM-based audio source separation inspired by the supervised approach in [60] but introducing the video information. The method in [60] needs to know *in advance* the sources that compose the mixture and their characteristics: the audio model for each source is built off-line. Here the information extracted from the video signal through the previous steps of our algorithm allows the application of the method without any off-line training. Thus, the separation that we perform is completely *blind* since no previous information about the sources is required.

The idea is to model the short time Fourier spectra of the sources by GMMs learned from training sequences $a_n^{train}(t)$. Using these models, the audio source separation is performed applying time-frequency masking on the Short Time Fourier Transform (STFT) domain. We will first explain our

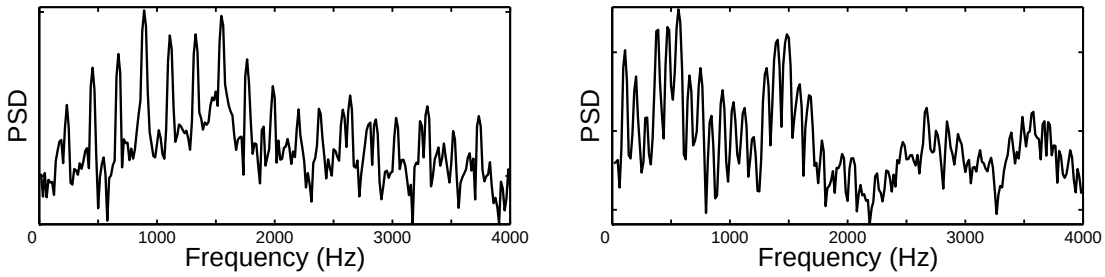


Figure 3.6 – *Example of* spectral *GMM states learned by our algorithm for female [left] and male [right] speakers. Each state* i *is represented by its PSD in dB:* $\log(\sigma_i^2(f))$.

model for the sources, next the process that we use to learn these models and finally the separation part.

Given an audio signal $a(t)$, we denote the STFT of this signal $A(t,f)$ and $A_{t'} = A(t,f)|_{t=t'}$ the short time Fourier spectrum of the signal at time t'. The short time Fourier spectra of the signal, A_t, are modeled with a GMM, i.e. the probability density function of A_t is given by

$$\mathrm{P}(A_t|\Lambda^{spec}) = \sum_i u_i N(A_t;\Sigma_i)\,, \tag{3.5}$$

with

$$N(A_t;\Sigma_i) = \prod_f \frac{1}{\pi\sigma_i^2(f)} \exp\left[-\frac{|A_t(f)|^2}{\sigma_i^2(f)}\right]\,. \tag{3.6}$$

Here $A_t(f)$ is the complex value of the short time Fourier spectrum A_t at frequency f and $\sigma_i^2(f)$, representing the local Power Spectral Density (PSD) at frequency f in the state i of the GMM, is the diagonal element of the diagonal covariance matrix $\Sigma_i = \mathrm{diag}[\sigma_i^2(f)]$. This *spectral* GMM is denoted $\Lambda^{spec} = \{u_i, \Sigma_i\}_i$.

Figure 3.6 shows two states of the GMMs that are learned by this method for a female [left] and a male [right] speaker. The states correctly characterize the sources frequency behavior: the male's audio energy is mainly present at lower frequencies (see Figure 3.6 [right]) while the female's harmonics (peaks in the PSD) start to appear at higher frequencies. A deeper analysis of this figure shows that for the female speaker, the fundamental frequency f_0 is around 220Hz (harmonics appear at multiples of 220Hz) while for the male it is around 110Hz. Those values for f_0 are within the range of the average speaking fundamental frequency for women (between 188 and 221 Hz) and for men (between 100 and 146 Hz) [5].

Let us now describe the **learning process**. For each source n, a training sequence $a_n^{train}(t)$ is composed of the detected time slots where the source is active alone, which are determined in Section 3.4.1. Next, the training sequence $a_n^{train}(t)$ is represented on the time-frequency plane $A_n^{train}(t,f)$ by applying a STFT using temporal windows of 512 samples length (64ms at 8kHz of sampling frequency) with 50% overlap. Then, the model $\Lambda_n^{spec} = \{u_{n,i}, \Sigma_{n,i}\}_i$ is learned by maximization of the likelihood $\mathrm{P}(A_{n_t}^{train}|\Lambda_n^{spec})$. This maximization is iteratively adjusted using the Expectation Maximization (EM) algorithm initialized by Vector Quantization (VQ) to Q_n states. The formulas used for the parameters re-estimation can be found in [8].

The method used for the **audio separation** is explained in Algorithm 1 for a mixture of $N = 2$ sources. This is done for simplicity and the procedure can be generalized to a higher number of sources. Thus, for each time instant we look for the most suitable couple of states given the mixture

Input: Mixture x, Spectral GMMs $\Lambda_n^{spec} = \{u_{n,i}, \Sigma_{n,i}\}_i$ and activity vectors w_n for the sources $n = 1, 2$

Output: Estimation of the sources audio part $\hat{a}_1$ and $\hat{a}_2$

A. Compute the STFT of the mixture $X(t, f)$ from the temporal signal x ;

foreach $t = 1, 2, \ldots, T$ **do**

1. Find the best combination of states according to the mixture spectrum X_t, that is

$$(i^*(t), j^*(t)) = \arg\max_{(i,j)} \gamma_{ij}(t)\,, \tag{3.7}$$

where $\gamma_{ij}(t)$ is the probability of choosing the combination of states (i, j) at time t for the observation X_t with $\sum_{ij} \gamma_{ij}(t) = 1$ and

$$\gamma_{ij}(t) \propto u_{1,i} u_{2,j} N(X_t; \Sigma_{1,i} + \Sigma_{2,j})\,. \tag{3.8}$$

2. Build a time-frequency local mask using knowledge about sources activity. For source $n = 1$:

$$\mathcal{M}_1(t, f) = \frac{\sigma^2_{1,i^*(t)}(f) \cdot w_1(t)}{\sigma^2_{1,i^*(t)}(f) \cdot w_1(t) + \sigma^2_{2,j^*(t)}(f) \cdot w_2(t)}\,, \tag{3.9}$$

and then $\mathcal{M}_2(t, f) = 1 - \mathcal{M}_1(t, f)$.

3. Apply the local masks to the mixture $X(t, f)$ to obtain the estimated source STFT:

$$\hat{A}_n(t, f) = \mathcal{M}_n(t, f) X(t, f)\,. \tag{3.10}$$

end

B. Reconstruct estimations of the sources audio part in the temporal domain $\hat{a}_n$ from the STFT estimations $\hat{A}_n$

Algorithm 1: Single-channel Audio Source Separation using knowledge about sources activity

spectrum. This information is used to build a time-frequency Wiener mask $\mathcal{M}$ for each source by combining the *spectral* PSDs in the corresponding states $(\sigma^2_{1,i^*(t)}, \sigma^2_{2,j^*(t)})$ with the knowledge about the sources activity w_n as explained in equation (3.9). When only one source is active, this weight w_n assigns all the soundtrack to this speaker. Otherwise, $w_n = 0.5$ and the analysis takes into account only the audio GMMs. In a further implementation we could assign intermediate values to w_n that account for the degree of correlation between audio and video. However, such cross-modal correlation has to be accurately estimated to avoid the introduction of separation errors.

3.5 BAVSS Performance Measures

3.5.1 Sources Activity Detection

The performance of the proposed method is highly related to accuracy in the estimation of the temporal periods in which each source is active *alone*. For our method, it is not fundamental to detect *all* the time instants during which sources are active alone, provided that the length of the detected period is long enough to train the source audio models. In fact, errors occur only when our algorithm estimates that only one source is active while in fact some of the other sources are active too. In these error frames our algorithm will learn an audio model for source S_i that represents

the frequency behavior of several sources mixed, and that will cause errors in the separation. Two measures assess the performance of our method in this domain: the *activity-error-rate* (ERR) and the *activity-efficiency-rate* (EFF).

Let N be the number of audio-visual sources and F_T be the number of video frames. For any fixed time and source S_n we define:

$$S_n^{\mathrm{ON}} := \text{“Source } S_n \text{ is active”}, \tag{3.11}$$

$$S_n^{\mathrm{OFF}} := \text{“Source } S_n \text{ is NOT active”}. \tag{3.12}$$

Let S_j with $j = 1, \ldots, N$, $n \neq j$ be the set of sources different from S_n. Then we define:

$$E_{j\neq n}^{\mathrm{OFF}} := \mathrm{AND}\,\{S_j^{\mathrm{OFF}}\ \forall j \neq n\}, \tag{3.13}$$

$$E_{j\neq n}^{\mathrm{ON}} := \mathrm{NOT}\,\{E_{j\neq n}^{\mathrm{OFF}}\} = \mathrm{OR}\,\{S_j^{\mathrm{ON}}\ \forall j \neq n\}. \tag{3.14}$$

$E_{j\neq n}^{\mathrm{OFF}}$ is the event where *all* sources different from S_n are inactive and $E_{j\neq n}^{\mathrm{ON}}$ is the complementary event where *one or more* of the sources different from source S_n are active.

The *activity-error-rate* (ERR) for source S_n is defined as

$$\underset{n}{\mathrm{ERR}} = \frac{F(S_n^{\mathrm{ON}}\ \mathrm{AND}\ E_{j\neq n}^{\mathrm{OFF}} | E_{j\neq n}^{\mathrm{ON}})}{F_T}, \tag{3.15}$$

where $F(B|C)$ is a function that returns the number of frames where our algorithm estimates that the event B has place and the ground truth soundtracks indicate that the current event is C. Thus, the ERR represents the percentage of time during which the algorithm makes an important error since it decides that only source S_n is active and it is not true (one or more of the other sources are active too).

The *activity-efficiency-rate* (EFF) for source S_n is defined as

$$\underset{n}{\mathrm{EFF}} = \frac{F(S_n^{\mathrm{ON}}\ \mathrm{AND}\ E_{j\neq n}^{\mathrm{OFF}} | S_n^{\mathrm{ON}}\ \mathrm{AND}\ E_{j\neq n}^{\mathrm{OFF}})}{F_n}, \tag{3.16}$$

where F_n is the number of frames where source S_n is active alone. Thus, the EFF represents the percentage of time in which a source is active alone that our method is able to detect. This parameter is very important given the short duration of the analyzed sequences: the higher is EFF, the longer is the period during which we learn the source audio models and, consequently, we can expect to obtain better results on the audio separation part.

3.5.2 Audio Source Separation

The BSS Evaluation Toolbox is used to evaluate the performance of the proposed method in the Audio Separation part. The estimated audio part of the sources $\hat{a}_n$ is decomposed into: $\hat{a}_n = a_{\mathrm{target}} + e_{\mathrm{interf}} + e_{\mathrm{artif}}$, as described in [80]. a_{target} is the target audio part of the source and e_{interf} and e_{artif} are, respectively, the interferences and artifacts error terms. These three terms should represent the part of $\hat{a}_n$ perceived as coming from the wanted source a_n, from other unwanted sources $(a_j)_{j\neq n}$ and from other causes. Two quantities are computed using this toolbox, the source-to-interferences ratio (SIR), and the sources-to-artifacts ratio (SAR), defined as:

$$\mathrm{SIR} = 10\log_{10}\frac{\|a_{\mathrm{target}}\|^2}{\|e_{\mathrm{interf}}\|^2} \tag{3.17}$$

$$\mathrm{SAR} = 10\log_{10}\frac{\|a_{\mathrm{target}} + e_{\mathrm{interf}}\|^2}{\|e_{\mathrm{artif}}\|^2} \tag{3.18}$$

Thus, the SIR measures the performance of our method in the rejection of the interferences and the SAR quantifies the presence of distortions and "burbling" artifacts on the separated audio sources. By combining SIR and SAR one can be sure of eliminating the interfering source without introducing too many artifacts in the separated soundtracks.

For a given mixture and using the knowledge about the original audio part of the sources a_n, oracle estimators for single-channel source separation by time-frequency masking are computed using the BSS Oracle Toolbox [81]. These oracle estimators are computed using the ground truth waveforms in order to result in the smallest possible distortion. As a result, SIR_{oracle} and SAR_{oracle} establish the upper bounds for the proposed performance measures. For further details about the oracles estimation, please refer to [81].

Finally, in order to compare our results to those obtained in [6], we compute the preserved-signal-ratio (PSR) for source S_n using the method described in [87] as

$$\underset{n}{\text{PSR}} = \frac{\|\mathcal{M}_n(t,f)a_n(t,f)\|^2}{\|a_n(t,f)\|^2}\,, \tag{3.19}$$

where $a_n(t,f)$ is STFT of the original audio signal corresponding to source S_n and $\mathcal{M}_n(t,f)$ is the time-frequency mask estimated using equation (3.9) and used in the audio demixing process. Thus, this measure represents the amount of acoustic energy that is preserved after the separation process.

3.6 Experiments

In a first set of experiments (Section 3.6.1), the proposed BAVSS algorithm is evaluated on synthesized audio-visual mixtures composed of two persons speaking in front of a camera. These sequences present an artificial mixture generated by temporally shifting the audio and video signals corresponding to one of the speakers so that it overlaps with the speech of the other person. The performance of the proposed method in identifying the number of sources in the scene, locating them the image and determining the activity periods of each one of them is assessed. Furthermore, a *quantitative* evaluation of the algorithm's results in terms of audio separation is performed since the original soundtracks (ground truth) of each speaker separately are available for these sequences.

As explained before, at present only two other methods have attempted a complete audio-visual source separation [6, 73]. The method presented in [73] does not provide any qualitative or quantitative result in terms of audio separation. In fact, this paper is mostly concentrated in the localization of the sources in the image and the only reference to the audio separation part states that the quality of the separated soundtracks is not good. Regarding the method presented in [6], two measures are used to evaluate quantitatively its performance in the audio separation part: the improvement of the signal-to-interference ratio (SIR) and the preserved-signal-ratio (PSR). In the last part of Section 3.6.1 these two quantities are used to compare our results to those obtained by the approach in [6] when analyzing sequences composed of two speakers.

In Section 3.6.2 we present a second set of experiments in which speakers and music instruments are mixed. The complexity of the sequences is higher given the more realistic background and the presence of distracting motion. These sequences are real audio-visual mixtures where both sources are recorded at the same time. Thus, it is not possible to obtain a quantitative evaluation of the algorithm's performances as in Section 3.6.1 since the audio ground truth is not available in this case. The main objective of Section 3.6.2 is to demonstrate *qualitatively* that our BAVSS method can deal successfully with complex real-world sequences involving speech and music instruments.

Videos showing all the experiments and the estimated audio-visual sources after applying our

method are available online at `http://lts2www.epfl.ch/~llagoste/BAVSSresults.htm`.

Let us now briefly summarize the main parameters in our approach.

- The number of audio and video atoms in which the sequence is decomposed: K and M respectively. In the experiments these parameters are fixed according to the characteristics of the signals, e.g. more audio atoms are used in Section 3.6.2 since the soundtracks are longer. However, the number of atoms extracted from the decomposition does not need to be set a priori, it can be automatically chosen setting a threshold on the reconstruction quality. Increasing the number of atoms would lead to a higher computational cost, but the results would not be significantly affected since the new atoms capture minor structures in the signals (due to the Matching Pursuit scalability explained in Section 2.2).

- The parameter that models delays between audio and video *relevant events* W in the atomic fusion method in Section 2.4. In all experiments $W = 13$ samples (0.45 seconds approximately), a time delay between a movement and the presence of the corresponding sound that appears to be appropriate. If we fix $W = 0$ our algorithm becomes more strict and we accept only events that are exactly synchronous. The more we increase W, the more tolerant we are and events need to be only approximately synchronous, which is more real. However, there is an intuitive maximum for W. Here we do not consider a video motion to be related to a sound if it occurs 0.45 seconds before that sound, because both events are too separated in time.

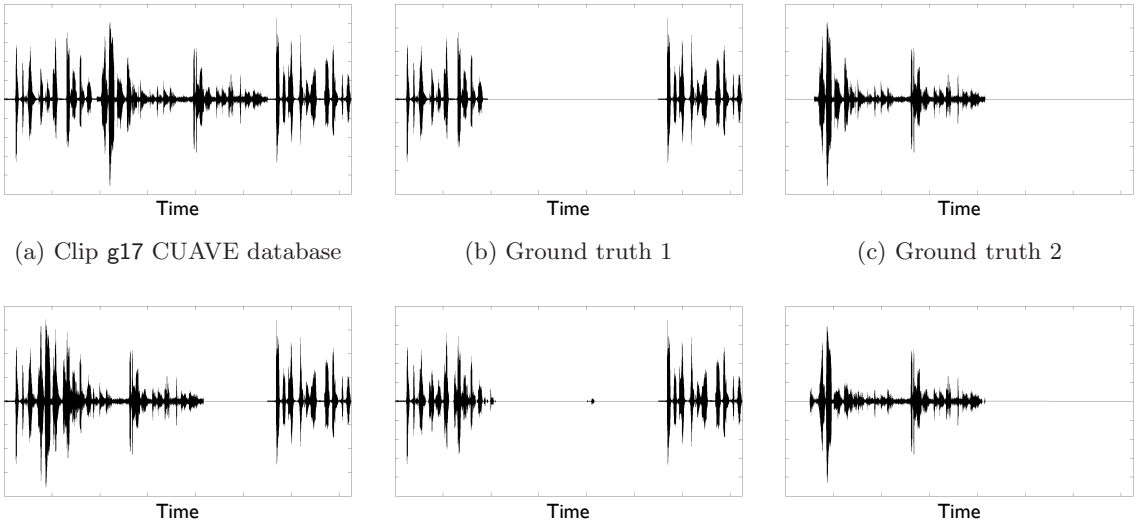
- The cluster size R that is used to group the video atoms and count the sources in the scene. In all experiments we use $R = 60$ pixels. The choice of this parameter is more critical and it needs to be performed according to the image dimensions. However, in the clustering algorithm in Section 3.3.1 we introduced a final step which is able to detect and eliminate *unreliable* clusters, i.e. clusters whose atoms present a *very low* coherence with the soundtrack. The purpose of this step is to increase the range of possible values for R. Thus, for R ranging between 40 and 90 pixels our clustering algorithm has been proved to detect the correct number of sources.

- The time period Δ that is used to determine if only one source is active in Section 3.4.1. We require all audio atoms in a period longer than Δ to be classified into the same source. In all experiments we fix Δ to 1 second (the sequences length is around 20 seconds or more). This value is small enough to ensure that *only one* source is active in this period, and big enough to train the source audio models. In fact, Δ can be set automatically according to the length of the analyzed clip, e.g. one tenth of the sequence length.

3.6.1 CUAVE Database: Quantitative Results

Sequences are synthesized using clips taken from the *groups* partition of the CUAVE database [62] with two speakers uttering sequences of digits alternatively. A typical sequence is shown in Figure 3.1. The video data is sampled at 29.97 frames/sec with a resolution of 480×720 pixels, and the audio at 44 kHz. The video has been resized to 120×176 pixels, while the audio has been subsampled to 8 kHz. The video signal is decomposed into $M = 100$ video atoms and the soundtrack is decomposed into $K = 2000$ audio atoms.

Ground truth mixtures are obtained by temporally shifting audio and video signals of one speaker in order to obtain time slots with both speakers active simultaneously. In the resulting synthetic clips, four cases are represented: both persons speak at the same time, only the boy or the girl speaks or silence. For further details on the procedure adopted to build the synthetic sequences the reader is referred to [42]. An example of this procedure on the audio part is shown on Figure 3.7. In (a) the



(a) Clip g17 CUAVE database (b) Ground truth 1 (c) Ground truth 2

(d) Mixture g17 (e) Estimated soundtrack 1 (f) Estimated soundtrack 2

Figure 3.7 – *Comparison between real (b)-(c) and estimated (e)-(f) soundtracks when analyzing a synthetic sequence (d) generated by applying a temporal shift to speaker 2 in clip g17 of CUAVE database (a).*

figure shows the original clip g17 of CUAVE database, in (b) the ground truth for source 1 (which is the period during which speaker 1 is uttering numbers) and in (c) the ground truth for source 2 which is obtained by shifting its audio part. In Figure 3.7 (d) we can see the input to our algorithm, a mixture built by adding ground truth waveforms 1 and 2. Figure 3.7 also gives a first *qualitative* evaluation of our method. It is possible to compare the ground truth to the estimated audio part of the sources separated using the proposed method (see Figure 3.7 (e)-(f)). Waveforms are very similar and the audible quality of the estimated sequences is also remarkable. The separation of the mixture when both sources are active is good as the numbers that each speaker is uttering are clearly understandable at a good quality.

Results obtained when analyzing ten different synthesized audio-visual sequences from CUAVE database are summarized in Table 3.1. In all cases the number of sources present in the scene and their position in the image has been correctly detected. Furthermore, the estimated position of the video source is always over the video part of the source and never over the background or the other source.

As explained before, two measures are used to evaluate the performance of our method in determining the time slots where sources are active alone. Results in Table 3.1 show that in all sequences the error rate (ERR) is under the 10%, and only in four cases we are over the 3%. Errors are concentrated in the boundaries of the source activity, that is just before the person starts to speak or after he/she stops, because in general motion in the video signal is not completely synchronous with sounds in the audio channel. Concerning our method's efficiency (EFF), only in three cases we are able to detect less than 50% of periods where sources are alone, and we average a 69%, which is a high percentage if we think about longer sequences. Low values for EFF are caused by the presence of video motion correlated to the audio on the source that is not active. In fact, it is difficult to detect the complete periods when sources are active alone without introducing errors, since there is a trade-off between them. If we choose to detect all the periods (EFF increases), more false positives will appear (ERR increases too) and, as explained before, the models for each source will not be correct. Here we prefer to have a high confidence when we decide that one source is

Sequence	Source	Activity accuracy (%)		SIR (dB)		SAR (dB)		PSR (%)
		ERR	EFF	blind	oracle	blind	oracle	
g12	$n=1$	0	74	14	33	4	19	83
	$n=2$	2	87	8	32	7	19	92
g13	$n=1$	3	64	10	36	4	21	66
	$n=2$	0	63	11	37	5	21	87
g14*	$n=1$	6	95	13	39	9	24	100
	$n=2$	0	73	25	39	4	22	65
g15	$n=1$	3	68					
	$n=2$	0	0					
g16	$n=1$	8	45	10	37	7	22	100
	$n=2$	2	82	18	38	3	21	56
g17	$n=1$	1	95	20	40	11	23	95
	$n=2$	0	83	29	39	11	24	94
g18	$n=1$	0	52	24	38	6	23	84
	$n=2$	10	69	12	38	7	22	94
g19*	$n=1$	6	44	15	33	7	19	86
	$n=2$	0	52	15	32	5	18	84
g20	$n=1$	0	90	20	35	9	21	88
	$n=2$	0	77	19	36	9	21	86
g21	$n=1$	0	64	16	38	6	23	87
	$n=2$	1	100	13	38	7	23	90
MEAN		2	69	16	37	7	21	85

Table 3.1 – *Results obtained with synthetic sequences generated for different clips of CUAVE database. Sequences marked with an asterisk (*) present two male speakers instead of one male and one female. Columns 1 and 2 represent respectively the analyzed sequence and the number of detected audio-visual sources. In Column 3 two quantities that evaluate the accuracy of our method in detecting the periods in which sources are active alone: the error rate [left] and the efficiency rate [right]. Columns 4 and 5 show a quantitative comparison between results on audio separation obtained using our blind method [left] and oracles computed using ground truth soundtracks [right]. Column 6 presents the percentage of energy from the original soundtrack that is kept after the audio separation process.*

active alone, even if then the efficiency decreases.

A 100% on EFF means that periods in which the source is active alone are perfectly detected. In this case, *blind* results for SIR and SAR are the best results that we can achieve using the GMM-based audio separation method in Section 3.4.2 since the training sequences are as long as possible. Consequently, the upper bounds for the performance in the blind separation of the audio track are clearly conditioned by the duration of the training sequences and the algorithm we use for the one microphone audio separation. While in some sequences the GMM-based separation seems suitable with performances up to 29dB of SIR (sequence g17), for some speakers this does not seem to be the case (8dB of SIR in sequence g12 even if the combined EFF for both speakers is 81%). However, taking into account the short duration of the analyzed sequences (20-30 seconds) and the training sequences (less than 8 seconds), results are satisfactory. Remember that the oracles in Table 3.1 represent the best results that we can obtain through *any* audio source separation method based on frequency masking *if we know in advance the ground truth soundtracks*. In fact, oracles guarantee the minimum distortion by computing the optimal time-frequency mask given the original separated

soundtracks. The average SIR that we obtain (16dB) is slightly better than the state-of-the-art on single-channel audio separation [8] and, unlike this method, we do it without any kind of supervision. As explained before, the joint processing of audio and video signals in our approach eliminates the necessity of knowing *in advance* the sources in the mixture and its acoustic characteristics, which is typical in one microphone audio separation methods. Furthermore, in all the resulting separated soundtracks here, even the ones that present worse SIR, the numbers that each speaker utters can be well understood.

In sequence g15 we can observe a major problem: there is no detected period when speaker 2 is active alone (see EFF in Table 3.1). Consequently, it is not possible to train a model of that source and our separation method cannot be applied. This happens because there is video motion correlated to the audio on source 1 (which is inactive) all over the duration of the period during which only source 2 is active. However, we can expect that with longer sequences (and longer time slots with each source active alone) this problem does not appear anymore, since in that case it is unlikely that correlated video motion is present on the inactive source all the time.

The audio separation task is extremely challenging for sequences g14 and g19, since in this case the mixture is composed by two male speakers. The fundamental frequencies of the speakers are extremely close and, as a result, their formants energy is highly overlapped in the spectrogram. Even in this difficult context, quantitative results (with an average SIR of 17dB) are close to those obtained when analyzing sequences with a male-female combination.

The comparison between our method and the approach in [6] presents some difficulties. First, the test set in [6] is composed of three very short sequences (duration ranging between 5 and 10 seconds), and only one of those sequences contains a mixture composed of speakers. Furthermore, they avoid distracting motion by locating the camera close to the speakers faces, i.e. we can only observe the lips in the video corresponding to the male speaker. Although the differences are considerable, here we compare the results in the speakers sequence in [6] with the mean results through all the sequences that we have analyzed. In [6], they report an improvement in the SIR of 14dB and a PSR of 57.5% (those values represent the mean between the male and female results). Here we obtain an average SIR of 16dB and an average PSR of 85%. Thus, our approach compares specially favorable in terms of PSR, that is the amount of acoustic energy that is preserved after the separation process. In fact, when demixing the audio part of the sources our methods keeps the 85% of the energy in the original audio signal while in [6] more than the 40% of this energy is lost. These results are related to the audio separation method used in each case: our GMM-based separation seems more suitable than the frequency tracking used in [6] when we consider the PSR.

3.6.2 LTS Database: Qualitative Results in a Challenging Environment

More challenging sequences including speakers and music instruments have been recorded in order to *qualitatively* test the performance of the proposed method when dealing with complex situations. The original video data is sampled at 30 frames/sec with a resolution of 240×320 pixels, and the audio at 44 kHz. For its analysis, the video has been resized to 120×160 pixels, while the audio has been sub-sampled to 8 kHz. The length of the sequences is close to 1 minute in this case. The video signal is decomposed into $M = 120$ atoms and the soundtrack is decomposed into $K = 6000$ atoms. As explained before, a quantitative evaluation can not be performed in this case since in this section we consider real mixtures where both sources are recorded at the same time.

In the first experiment (Movie1) we analyze an audio-visual sequence where two persons are playing music instruments in front of a camera. A frame of this movie is shown in Figure 3.8. In

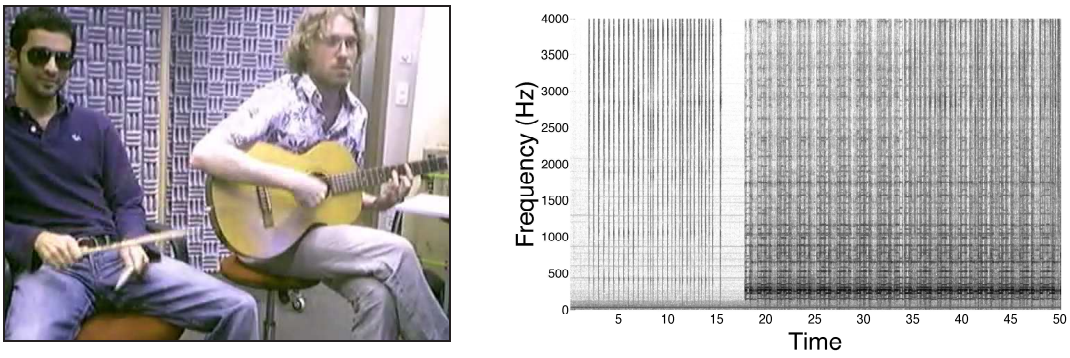


Figure 3.8 – *Challenging audio-visual sequence where one person is playing a guitar and another one is hitting two drumsticks in a complex background. A frame of this movie [left] and the corresponding audio spectrogram [right] are represented. Drumsticks are active in the begining of the sequence, then the guitarist starts to play and finally both instruments are mixed.*

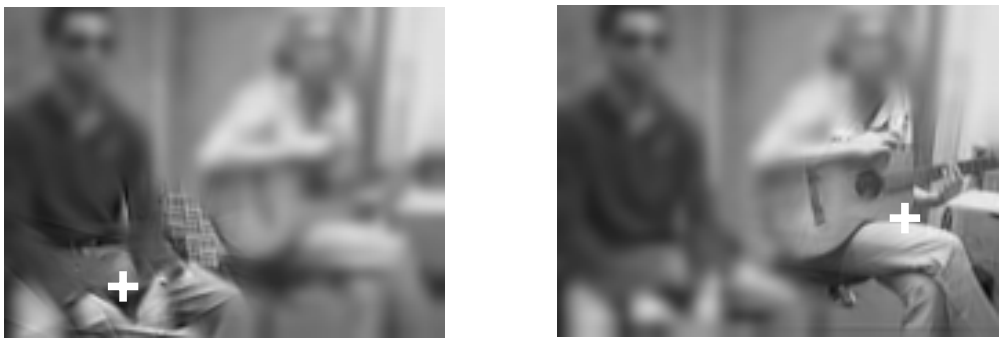


Figure 3.9 – *Video sources reconstruction for Movie1. The atoms that are highlighted in the images are those that characterize the left source [left] and the right source [right] respectively. Background is composed of the residual energy after the 3D-MP video decomposition and provides an easier visualization of the reconstructed sources. Finally, crosses mark the position in the image where our algorithm locates the sources.*

some temporal periods they play at the same time while in others they do a *solo*. A first difficulty is given by the fact that the video decomposition has to reflect the movement of the present structures, which is not an easy task when trying to model the drumsticks and their trajectory. Thus, while the hand that is playing the guitar moves in a smooth way, drumsticks movement is much more fast and abrupt. Another problem are some movements correlated with the sound, specially those of the guitarist's leg, and the proximity of the sources. If we compare this sequence with the ones presented in the literature we can see that, in those cases, either the sources are much more separated in the image [73] or distracting motion is avoided by visually zooming into the sources [6]. Furthermore, these methods always present flat, or almost flat, backgrounds. Here the complex background (see Figure 3.8) makes the video decomposition task more complicated since a considerable part of the video atoms has to be used to represent it.

When analyzing Movie1 with the proposed BAVSS method, the number of sources and its position in the image are perfectly detected (see crosses on Figure 3.9). A reconstruction of the image using the atoms assigned to each source is shown in Figure 3.9. In the left picture it is possible to see how the stick is successfully represented by one video atom, and in the right one, the atoms that surround the guitar are highlighted. In this sequence, the activity periods of each source are also

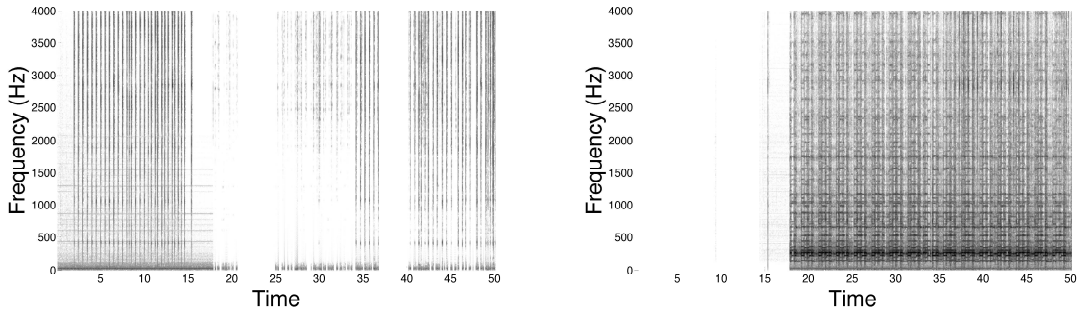


Figure 3.10 – *Estimated spectrograms for drumsticks [left] and guitar [right] in Movie1. Drumsticks are silent in the middle of the sequence and the guitar at the beginning. Spectrograms show that the sources behavior is correctly detected by the proposed method.*

(a) (b)

Figure 3.11 – *Two frames belonging to Movie2 (a) and Movie3 (b). On both frames, one person is uttering numbers while a guitarist is playing. Frame (b) shows the distracting motion caused by a person who is crossing the scene behind the sources. The estimated source positions are marked with crosses.*

detected. A good characterization of the sources in the frequency domain is achieved, which leads to a satisfactory audio separation of the sources. Figure 3.10 shows the spectrograms that we obtain. We can see that drumsticks sounds [left] are much more sharp in the spectrogram (well-localized in time, broad range in frequency) while the guitar spectrogram [right] has much more energy and it is composed by several harmonic sounds. Concerning the audible quality of the estimated soundtracks, the audio part of the drumsticks is perfectly reconstructed at the beginning and it only presents some distortion at the end, where they are mixed with the guitar sounds. In addition, it is almost impossible to hear the guitar in the drumsticks soundtrack. Finally, the quality of the guitar reconstruction is good even though there are some attenuated drumstick sounds in the last part.

Second and third experiments are very similar. They present an audio-visual mixture composed of speech and guitar sounds. In Movie2 a male speaker is uttering numbers (see Figure 3.11(a)), while in Movie3 there is a female speaker and another person crosses the scene generating thus distracting motion (Figure 3.11(b)). These sequences share one challenging difficulty, the fact that acoustic energy of the guitar is considerably stronger than the energy coming from the speech. Furthermore, it is not possible to equalize the energies of both sources since they are recorded at the same time.

Results obtained when analyzing these two sequences are similar. The number of present sources and their spatial position are correctly determined (see crosses in Figure 3.11). Despite of not detecting the whole periods during which each source is active alone, the periods that we detect are

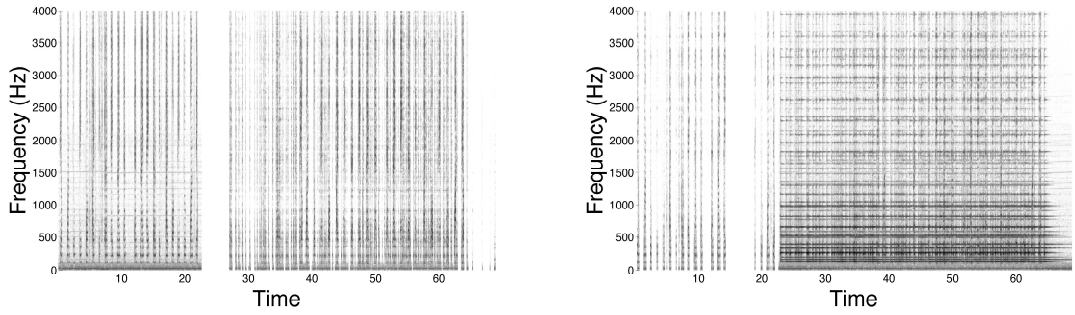


Figure 3.12 – *Estimated spectrograms for speech [left] and guitar [right] in Movie2. In the first part the speaker is uttering numbers alone, next there is a short period where the guitar starts to play while the speaker is silent and in the last part both sources are mixed.*

correct and long enough to represent the sources frequency behavior. Finally, concerning the audio separation part, even though the speakers estimated soundtracks are pretty clean, in the case of the guitar we can still hear speech. A first reason for this behavior is the unbalanced energy between sources that we discussed before. Another one, and maybe the main one, could be the fact that the guitar sounds present many harmonics that overlap with speech in the spectrogram. Thus, some frequency formants of speech are also characterized in the acoustic model of the guitar and we can not eliminate them in the audio separation part using this separation method.

Spectrograms of the estimated audio part of the sources for Movie2 can be observed in Figure 3.12. We can observe that the short time slot where the guitar is active alone is perfectly detected (between seconds 22 and 27) since it is not present in the speaker spectrogram [left]. It is also possible to see the residual energy of the speech signal that remains in the first part of the guitar spectrogram [right].

Even if the distracting motion present on Movie3 (see Figure 3.11(b)) seems not to affect the performance of the proposed method, results concerning the audio separation are slightly worse in this case. However, since the activity periods for the sources are also correctly detected, this degradation in performance cannot be due to the background motion but rather to the fact that female harmonics overlap more often with the guitar ones in the spectrogram.

3.7 Discussion

In this chapter we have introduced a novel algorithm to perform Blind Audio-Visual Source Separation. We consider sequences made of one audio signal and the associated video signal, without the stereo audio track usually employed for the audio source separation task. The method correlates salient acoustic and visual structures that are represented using atoms taken from redundant dictionaries. Video atoms synchronous with the soundtrack are grouped together using a clustering algorithm that counts and localizes on the image plane audio-visual sources. Then, using this information and exploiting the coherence between audio and video signals, the audio activity of the sources is determined and its audio part is separated and reconstructed.

One of the contributions of this work is an extensive evaluation of the proposed method on sequences involving speakers and music instruments. This systematic study of the algorithm performances represents a sensible improvement with respect to previously published works in [6, 73] that test algorithms' performances on few, very short sequences. Here, a first set of experiments

has been performed on synthetic sequences built from CUAVE database in which two persons utter numbers in front of a camera. In all cases, the scene has been well interpreted by our algorithm, leading to state-of-the-art audio-visual source separation. The audible quality of the separated audio signals is good. A rigorous evaluation of the audio separation results has been performed using the BSS Evaluation Toolbox. These quantitative results do not show any significant difference between sequences where two male speakers are mixed and those where a male and a female appear. A second set of tests has been performed on more realistic sequences where speakers are mixed with music instruments. Even if the nature of this second set of sequences does not allow a quantitative evaluation of the results, we have demonstrated that the proposed BAVSS method is able to deal with less static sources, complex backgrounds and distracting motion representing a much more realistic environment. Given the short length of the analyzed sequences, a possible improvement for the audio separation part could be the adaptation of a general acoustic model to the detected sources as explained in [60].

Joint Audio-Visual Processing using Nonlinear Diffusion

4

4.1 Motivation

Approaches in audio-visual analysis try to assess the synchrony between audio and video channels in order to extract information about the observed scene. Thus, in most applications only the video parts that are related to the soundtrack are used. For example, speech recognition only needs the region around the mouth, and approaches in sound source localization search for regions moving coherently with the sounds. The remaining video information, such as background and video structures whose motion is not related to the soundtrack, is superfluous and not necessary for those audio-visual applications. However, identifying a mouth or discriminating audio-related motion from purely distracting motion involves a significant amount of computational cost. In this chapter we aim at simplifying audio-visual sequences by eliminating most of this non-relevant video information through a computationally inexpensive procedure.

After Perona and Malik's preliminary work in [64], nonlinear diffusion has been proven a useful tool for the *selective* removal of information in a given signal. This technique has been successfully applied to image denoising, restoration and edge detection [3, 15, 56, 85]. Furthermore, the flexibility in the design of the diffusion coefficient (which controls the intensity of the diffusion at each point of the signal) makes it applicable to a great variety of problems. As explained before, our work seeks the elimination of the video information which is not related to the soundtrack. Thus, we can simply integrate the basic assumption of audio-visual analysis in the definition of the diffusion coefficient. Here we propose a nonlinear video diffusion approach which is controlled by a diffusion coefficient that is a function of the synchrony between audio energy and video motion at each point of the video domain. Our diffusion model is inspired by the variant of the classic Perona-Malik model [64] that Catté et al. proposed in [15]. This nonlinear diffusion approach based on partial differential equations (PDEs) has been demonstrated to provide good results in the above mentioned applications. Our method is designed to remove the information in parts of the video signal whose

The work presented in this chapter has been submitted for publication in [47]. A preliminary work on this subject can also be found in [46].

motion is not coherent with a synchronously recorded audio track, while preserving *regions* that are useful for applications in the audio-visual domain. And this is another reason to use diffusion: we want to favor regions instead of pixels. The 3D diffusion process brings implicitly the spatio-temporal coherence that we seek, since the diffusion is a smooth way to transport mass (grey value in our case).

The main contribution of the proposed approach is the definition of this audio-visual diffusion coefficient. In this chapter we also introduce a measure that is able to quantify the efficiency of our approach by comparing the strength of the diffusion inside and outside the audio-related video regions. After this measure is defined, we are able to discuss appropriate values for the main parameter in our approach based on its effect on our method's efficiency. We also present a discretization scheme that ensures the diffusion process stability and prevents the creation of new maxima. Furthermore, we propose a stopping criterion that is appropriate for our objectives, and at the same time intuitive and computationally inexpensive. The proposed approach is demonstrated on challenging real-world sequences, all of them presenting important auditive and/or visual distractors. Several experiments illustrate the robustness of our method to the presence of Gaussian noise in the input signals.

The chapter is structured as follows. In Section 4.2 the main principles of PDE-based diffusion are recalled and the problem is formally defined in the continuous domain. Section 4.3 presents the proposed model for audio-based nonlinear video diffusion by detailing the composition of the audio-visual diffusion coefficient. Section 4.4 details the numerical scheme used for the problem discretization, which ensures the stability of our diffusion procedure. In Section 4.5 a quantitative measure of our method's efficiency is introduced and used to discuss appropriate values for the main parameter in our approach. In Section 4.6 we define the stopping criterion for the audio-visual diffusion process. Section 4.7 presents the results when analyzing challenging natural audio-visual sequences and our method's behavior when the signals are corrupted by white noise. Finally, in Section 4.8 achievements and future research directions are discussed.

4.2 PDE-based Diffusion

The diffusion equation can be derived from the *continuity equation*, which states that a change in density in any part of a system is due to inflow and outflow of material into and out of that part of the system:

$$\partial_\tau v = -\operatorname{div} \mathbf{j}\,, \tag{4.1}$$

where $\mathbf{j}$ is the flux of the diffusing material and τ denotes the diffusion time. Thus, the diffusion process does only transport mass (or grey value in the case of image processing) without destroying or creating new mass. Then, the diffusion equation can be obtained when combining this continuity equation with *Fick's first law*, which assumes that the flux $\mathbf{j}$ of the diffusing material in any part of the system is proportional to the local density gradient ∇v as

$$\mathbf{j} = -D \cdot \nabla v\,. \tag{4.2}$$

The relation between ∇v and $\mathbf{j}$ is described by a *diffusion coefficient* D.

Let us consider a 3D video domain $\Omega := (0, b_1) \times (0, b_2) \times (0, b_3)$ with boundary $\Gamma := \partial\Omega$ and let a video signal v be represented by a mapping $f \in \mathrm{L}^\infty(\Omega)$. Then, a general continuous model for

anisotropic diffusion filters is represented by the following boundary value problem:

$$\partial_\tau v = \mathrm{div}(D\nabla v) \quad \text{on} \quad \Omega \times (0,\infty)\,, \tag{4.3}$$

$$v(\mathbf{x},0) = f(\mathbf{x}) \quad \text{on} \quad \Omega\,, \tag{4.4}$$

$$\langle D\nabla v, \mathbf{n}\rangle = 0 \quad \text{on} \quad \Gamma \times (0,\infty)\,. \tag{4.5}$$

Here $\mathbf{n}$ denotes the outer normal, $\mathbf{x} = (x, y, t)$ are the 3D video coordinates and $\langle\cdot,\cdot\rangle$ is the Euclidean scalar product on $\mathbb{R}^3$. In equation (4.3), $\mathrm{div}(\cdot)$ and ∇ denote respectively the divergence and the gradient operators with respect to the space variables. Notice that τ is used for the diffusion time and t for the temporal axis of the video signal.

The diffusion equation in (4.3) belongs to a general class of equations satisfying the *maximum principle*. The principle states that all the maxima of a solution of equation (4.3) for diffusion times $\tau \in [\tau_0, \tau_1]$ are to be found on the boundary Γ or at $\tau = \tau_0$ provided that the diffusion coefficient D is positive. Since our boundary problem is also composed of equation (4.5), the diffusion is 0 across the boundary Γ and the maxima can only belong to the original image (initial condition at $\tau = \tau_0$). A proof of the maximum principle can be found in [57]. In practice, this is a very important property since the principle prevents the creation of new local extrema when applying the diffusion process to any function v.

Chronologically, applications in the signal processing domain have evolved from using simple constant values for the diffusion coefficient D until much more complex expressions. We can briefly summarize the process by citing the most representative steps of this evolution according to the diffusion coefficient properties.

Linear diffusion: The diffusion coefficient is constant on space and diffusion time: $D(\mathbf{x}, \tau) = c$, where c is a scalar. This is the simplest and most studied case. The solution of the resulting diffusion equation $\partial_\tau v = c\Delta v$ is equivalent to convolving the original video signal $v(\mathbf{x}, 0)$ with a Gaussian of variance $\sigma = \sqrt{2\tau}$ (see [85] for a more detailed explanation). Thus, the effect of applying linear diffusion to a given signal is the Gaussian blurring of this signal.

Scalar-valued nonlinear diffusion: At each point $\mathbf{x}$ and iteration step τ the diffusion coefficient is represented by a scalar value: $D(\mathbf{x}, \tau) \in \mathbb{R}$, $\forall\, \mathbf{x}, \tau$. In this case, the diffusion process is called inhomogeneous and nonlinear since it depends on the spatial coordinate $\mathbf{x}$ and the iteration τ respectively. In fact, the diffusion coefficient depends on the evolving video signal itself. This model was first proposed by Perona and Malik in [64] and it is commonly applied to edge detection.

Vector-valued nonlinear diffusion: The diffusion process is controlled by a tensor that depends on the spatial coordinate $\mathbf{x}$ and the iteration step τ: $\mathbf{D}(\mathbf{x}, \tau) \in \mathbb{R}^{3\times3}$, $\forall\, \mathbf{x}, \tau$. This characteristic gives more freedom to the diffusion process and it can be applied to detection of corners or line-like structures. The approach proposed by Weickert in [85] belongs to this group.

For a deeper understanding of PDE-based diffusion, please refer to [3, 85].

4.3 Audio-based Video Diffusion

We seek to diffuse parts of a video signal whose temporal variations do not correlate with a synchronously recorded soundtrack, since this video information is not used in most applications in joint audio-visual processing. In this section we introduce an audio-visual diffusion coefficient D

which is able to achieve this purpose. The structure of the proposed coefficient assesses audio-video coherence and keeps only regions that are interesting for audio-visual analysis. For this purpose, we rely again on the assumption of synchrony between related events in audio and video channels.

We propose the following scalar-valued diffusion coefficient D:

$$D(\mathbf{x}, \tau) = g(|s_\sigma(\mathbf{x}, \tau)|^2)\,, \tag{4.6}$$

where $g(\cdot)$ is a function that determines the intensity of the diffusion process at each point of the video volume and s_σ is a regularized measure of the synchrony between audio and video channels which is defined as

$$s_\sigma(\mathbf{x}, \tau) = (a(\mathbf{x})\partial_t v(\mathbf{x}, \tau)) * G_\sigma(\mathbf{x})\,. \tag{4.7}$$

In this expression, G_σ is a 3D Gaussian of variance σ^2, $\partial_t v$ is the temporal derivative of the video signal, and $a(x, y, t) = a(t)\ \forall x, y$ represents the energy on the audio channel at time t (notice that the audio feature does not depend on the spatial coordinates x and y). Thus, the *audio-video synchrony* s_σ evaluates the coherence between both channels by combining audio energy and video motion at each point $\mathbf{x}$ of the video volume. According to the expression in (4.7), $|s_\sigma|$ is high when an important acoustic event matches a relevant pixel motion while its value is close to zero in the rest.

The convolution with a Gaussian G_σ in expression (4.7) makes our audio-visual synchrony measure s_σ much more robust to visual and acoustic noise and ensures spatio-temporal coherence to our method. Furthermore, this procedure has been used by Catté et al. in [15] in order to regularize the nonlinear diffusion problem presented by Perona and Malik in [64], whose formulation is similar to ours. In all experiments the regularization parameter is fixed to $\sigma = 1$. This value has been shown in [56] to be sufficient for a large interval of noise variances when the noise in neighboring pixels is uncorrelated and the grid size is one, which is true in our case (see equation (4.11) in Section 4.4).

Some considerations should be taken into account regarding the audio and video features that we use in equation (4.7) to estimate the audio-video synchrony s_σ and thus the diffusion coefficient D. As explained before, the audio feature $a(t)$ represents the energy in the audio channel and the video feature $\partial_t v$ corresponds to the motion in the video signal. However both features have been processed to improve the performance of the proposed method. Thus, the audio feature $a(t)$ is an *equalized* audio energy, while the video feature $\partial_t v$ is also an *equalized* video motion, which means that all the "peaks" in each domain have approximately the same magnitude. This is a very important point because it ensures that our approach will give the same opportunities to all the significant motion and sounds instead of keeping only the region presenting the most intense motion and occurring exactly at the same time as the louder sound. As a result, the movements that are related to the soundtrack can be effectively preserved even if they are significantly smaller than a distracting motion in the scene. Some examples of the original and the equalized audio and video features can be observed in Figure 4.1. The audio feature [bottom right] has approximately the same magnitude for all significant sounds recorded with the microphone even if originally they have very different energy. Regarding the video signal, the strong motion corresponding to a rocking horse and the mouth movements which are hardly visible in Figure 4.1 [center left] are also represented by a similar magnitude in the video feature [bottom left]. The *equalization* in audio and video domains is performed in the same way. First, we convolve the original signal with two Gaussians of different variances (a 3D Gaussian in the case of the video motion and a 1D Gaussian for the audio energy). Then, the equalized features are the quotient of dividing the result of the convolution with the thinner and thicker Gaussians respectively. Thus, each peak in audio energy and video motion is compared to the energy/motion in the region around it and both features become relative

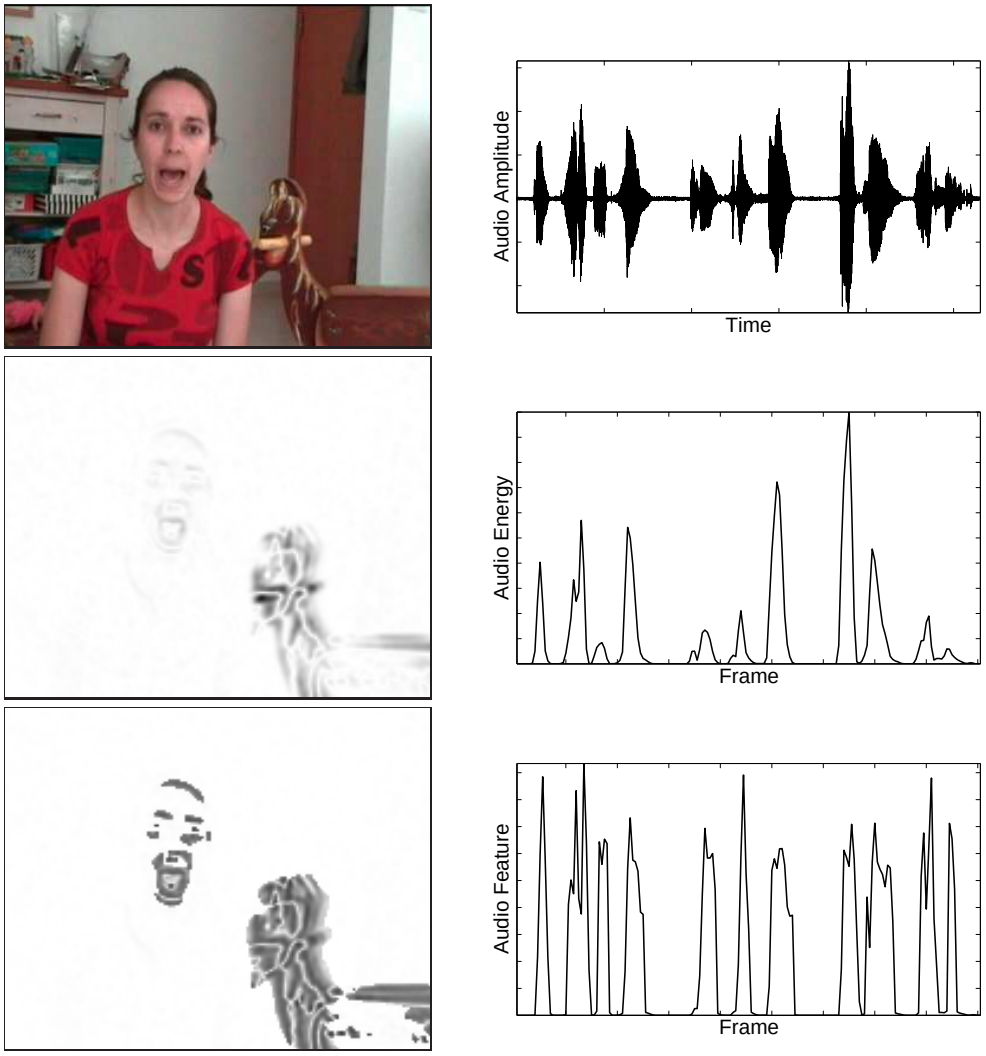


Figure 4.1 – *Proposed features [bottom] corresponding to the audio and video signals in the top row. Right column shows [from top to bottom] the original audio signal, its energy and the equalized energy $a(t)$ at the same temporal resolution than the video signal. Left column depicts one video frame, the motion in this frame (magnitude of the pixels' temporal variation) and the corresponding equalized motion, that is $\partial_t v(x, y, t)$ for a fixed time t. White regions represent static pixels.*

measures. Other features for audio and video signals could also be used. For example in the audio case we could use a smoothed version of a binary audio activity detector or the acoustic energy in an important audio sub-band. In any case the features should not be very selective since audio and video channels are never exactly synchronous.

Let us now discuss the shape of the function $g(\cdot)$ in equation (4.6). As discussed before, we want a linear diffusion process to take place in spatio-temporal regions with low audio-visual synchrony. In addition, the diffusion coefficient D should be close to 0 in points with high $|s_\sigma|$ in order to stop there the diffusion. Thus, $g(\cdot)$ should be a non-negative monotonically decreasing function with $g(0) = 1$, since the diffusion coefficient D needs to be positive. An appropriate shape for function

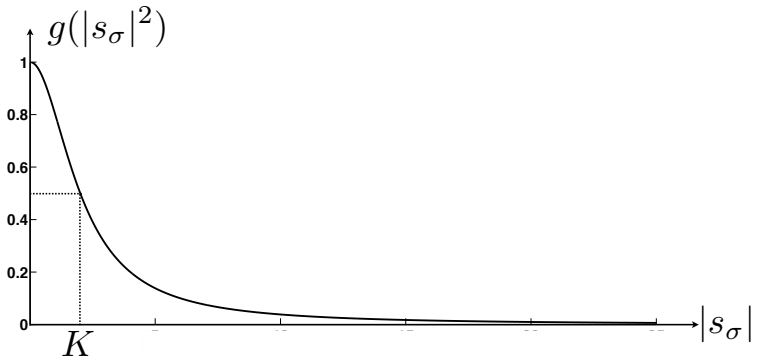


Figure 4.2 – *Shape of the function* $g(\cdot)$ *in equation* (4.8).

$g(\cdot)$ can then be the function proposed by Perona and Malik in [64] (see Figure 4.2):

$$g(|s_\sigma|^2) = \frac{1}{1 + \frac{|s_\sigma|^2}{K^2}} \ . \tag{4.8}$$

The value of the constant K should be chosen carefully since it acts as a threshold: points where $|s_\sigma| < K$ are strongly affected by linear diffusion (Gaussian blurring) while those points where $|s_\sigma| > K$ are least diffused. A deep discussion about the choice of an appropriate value for this constant and its effect on the efficiency of the proposed method is provided further in Section 4.5.

We can now analyze qualitatively the behavior of the proposed audio-visual diffusion process given the diffusion coefficient defined in equation (4.6). First of all, the diffusion coefficient is maximal and constant to $D(\mathbf{x}, \tau) = 1$ in video *regions* where $s_\sigma = 0$, that is:

1. Static video regions (video inactivity).
2. Silent time slots (audio inactivity).
3. Situations where the visual motion is not synchronous with the appearance of sounds (audio-video incoherence).

Since inside these regions, the diffusion coefficient is constant to 1, the diffusion equation in (4.3) becomes the heat equation ($\partial_\tau v = \Delta v$) and the region is diffused in an homogeneous way. Out of those regions, the diffusion coefficient D becomes smaller and the diffusion process is stopped. In fact, the larger is $|s_\sigma|$ the lower is the level of diffusion that a pixel experiences. In addition, the nature of linear 3D diffusion together with the regularization with a Gaussian G_σ in equation (4.7) bring implicitly spatial coherence to our approach by favoring structures over pixels. Notice that the diffusion coefficient $D \approx 1$ in a pixel that is surrounded by pixels with low audio-visual synchrony $|s_\sigma|$, independently of the synchrony of the pixel itself. Thus, only spatio-temporal *regions* whose movement is coherent with the soundtrack are preserved. This characteristic represents a significant advantage over other methods that are based on single pixel movement [6, 36, 58, 73, 75], since our approach is in consequence less vulnerable to noise.

As a summary, we are performing a *nonlinear* diffusion over a 3D volume (the video signal) which is controlled by a diffusion coefficient D that depends on the synchrony between audio and video signals. The proposed diffusion process leads to the blurring of the visual structures that are not relevant for joint audio-visual processing while it keeps a good resolution in the rest. Thus, the resulting video signal naturally highlights the possible sound sources in the scene.

4.4 Discretization

The continuous model for audio-visual diffusion has been defined in Sections 4.2 and 4.3. Here, we detail the discretization of the proposed approach by means of finite differences. Finite difference schemes are widely used in signal processing due to the structure of a digital signal as a set of uniformly distributed pixels. Thus, it is very natural to associate a video signal to a uniform 3D grid. The discretization scheme that we apply to the diffusion problem stated in equations (4.3)-(4.5) has been studied and advised by Aubert and Kornprobst in [3].

The continuous diffusion equation in (4.3) can be rewritten as

$$\partial_\tau v = \partial_x(D\partial_x v) + \partial_y(D\partial_y v) + \partial_t(D\partial_t v)\,. \tag{4.9}$$

The left part of this equation has been discretized following a *forward* finite difference scheme as commonly done in literature [3]:

$$\partial_\tau v|^n_{ijk} \approx \delta^+_\tau v^n_{i,j,k} := \frac{v^{n+1}_{i,j,k} - v^n_{i,j,k}}{\Delta\tau}\,, \tag{4.10}$$

where $v^n_{i,j,k}$ is the value of v at location $(i\Delta x, j\Delta y, k\Delta t)$ and diffusion time $n\Delta\tau$. Here Δx, Δy and Δt are the grid spacing used in the discretization of the video dimensions, while $\Delta\tau$ is the grid spacing used for the diffusion time discretization. In fact, the $\Delta\tau$ parameter controls the speed with which the diffusion process affects the video signal v. In this work, the pixel size is chosen as the unit of reference in all spatio-temporal dimensions:

$$\Delta x = \Delta y = \Delta t = h = 1\,. \tag{4.11}$$

Using the result in equation (4.10) and rearranging the terms, the continuous diffusion equation in (4.9) has been discretized using a finite differences scheme as

$$v^{n+1}_{i,j,k} = v^n_{i,j,k} + \Delta\tau\left(\delta^*_x(D^n_{i,j,k}\delta^*_x v^n_{i,j,k}) + \delta^*_y(D^n_{i,j,k}\delta^*_y v^n_{i,j,k}) + \delta^*_t(D^n_{i,j,k}\delta^*_t v^n_{i,j,k})\right)\,, \tag{4.12}$$

where

$$\begin{aligned}
\partial_x v|_{ijk} \approx \delta^*_x v_{i,j,k} &:= \frac{v_{i+\frac{1}{2},j,k} - v_{i-\frac{1}{2},j,k}}{\Delta x}\,,\\
\partial_y v|_{ijk} \approx \delta^*_y v_{i,j,k} &:= \frac{v_{i,j+\frac{1}{2},k} - v_{i,j-\frac{1}{2},k}}{\Delta y}\,,\\
\partial_t v|_{ijk} \approx \delta^*_t v_{i,j,k} &:= \frac{v_{i,j,k+\frac{1}{2}} - v_{i,j,k-\frac{1}{2}}}{\Delta t}\,,
\end{aligned} \tag{4.13}$$

are the discretizations of the derivatives in the x, y and t directions respectively. Then, the values of v at location $((i\pm\frac{1}{2})\Delta x, (j\pm\frac{1}{2})\Delta y, (k\pm\frac{1}{2})\Delta t)$ are obtained by linear interpolation. Notice that we can rewrite for example $\delta^*_t v_{i,j,k}$ as the common *centered* approximation of $\partial_t v|_{ijk}$:

$$\delta^*_t v_{i,j,k} = \frac{v_{i,j,k+1} - v_{i,j,k-1}}{2\Delta t}\,, \tag{4.14}$$

since the linear interpolation is defined as

$$v_{i,j,k+\frac{1}{2}} = \frac{v_{i,j,k+1} + v_{i,j,k}}{2}\,. \tag{4.15}$$

The same reasoning applies to the discretizations of the derivatives in the x, y directions.

Developing and rearranging the terms in equation (4.12) we obtain the following difference scheme:

$$v_{i,j,k}^{n+1} = v_{i,j,k}^{n} \left(1 - \frac{\Delta\tau}{h^2} \sum_l D_l^n\right) + \frac{\Delta\tau}{h^2} \sum_l D_l^n v_l^n \,, \tag{4.16}$$

where $l = \{E, W, N, S, F, R\}$ are the mnemonic subscripts for East, West, North, South, Front, Rear, and

$$\begin{aligned} D_E &= D_{i+\frac{1}{2},j,k} \,, & v_E &= v_{i+1,j,k} \,, \\ D_W &= D_{i-\frac{1}{2},j,k} \,, & v_W &= v_{i-1,j,k} \,, \\ D_N &= D_{i,j+\frac{1}{2},k} \,, & v_N &= v_{i,j+1,k} \,, \\ D_S &= D_{i,j-\frac{1}{2},k} \,, & v_S &= v_{i,j-1,k} \,, \\ D_F &= D_{i,j,k+\frac{1}{2}} \,, & v_F &= v_{i,j,k+1} \,, \\ D_R &= D_{i,j,k-\frac{1}{2}} \,, & v_R &= v_{i,j,k-1} \,. \end{aligned} \tag{4.17}$$

Thus, at each point $(i\Delta x, j\Delta y, k\Delta t)$ and iteration $n+1$ the intensity of the video signal depends only on its previous intensity and the intensities of the six closest spatial neighbors at iteration n. The contribution of each spatial neighbor v_l is determined by the interpolated diffusion coefficient D_l.

Concerning the rest of the studied boundary value problem, the initial condition in equation (4.4) has been discretized as

$$v_{i,j,k}^0 = f(i\Delta x, j\Delta y, k\Delta t) \,, \tag{4.18}$$

and the original video signal is used as initial condition. Finally, the boundary condition in equation (4.5) has been accomplished by setting the diffusion coefficient D to zero at the boundaries of the 3D video signal.

Let us now discuss the properties of this discretization scheme. In our approach, a choice of $\Delta\tau \in [0, 1/6]$ ensures the positiveness of all coefficients in equation (4.16) since $h = 1$ and $D \in [0, 1]$. Under those conditions, the proposed discretization satisfies the maximum and minimum principle, whose importance was explained in Section 4.2. This can be proven easily by extending from two to three dimensions the demonstration performed by Perona and Malik in [64]. Thus, if we define the maximum and the minimum of the neighbors of $v_{i,j,k}$ at iteration n as $v_M = \max\{(v, v_l)_{i,j,k}^n\}$ and $v_m = \min\{(v, v_l)_{i,j,k}^n\}$ for $l = \{E, W, N, S, F, R\}$, we can prove that:

$$(v_m)_{i,j,k}^n \leq v_{i,j,k}^{n+1} \leq (v_M)_{i,j,k}^n \ . \tag{4.19}$$

Assuming $\Delta\tau \in [0, 1/6]$, $D \in [0, 1]$ and $h = 1$ we can write from equation (4.16):

$$v_{i,j,k}^{n+1} \leq (v_M)_{i,j,k}^n (1 - r\Delta\tau) + (v_M)_{i,j,k}^n r\Delta\tau = (v_M)_{i,j,k}^n \tag{4.20}$$

$$v_{i,j,k}^{n+1} \geq (v_m)_{i,j,k}^n (1 - r\Delta\tau) + (v_m)_{i,j,k}^n r\Delta\tau = (v_m)_{i,j,k}^n \tag{4.21}$$

where $r = \sum_l D_l^n$. As a result, at each iteration the maximum and the minimum of v become closer and no new maxima or minima are created. Furthermore, this guarantees the stability of the proposed discretization scheme since it prevents the video pixels' intensity from growing in time.

4.5 Audio-Visual Diffusion Ratio α and Study of the Diffusion Parameter K

The proposed diffusion procedure seeks the elimination of video information that is not relevant for joint audio-visual processing. The criterion used to determine if a certain part is relevant is

the synchrony between video motion and audio energy. Video parts whose motion is not coherent with the audio channel activity are affected by homogeneous diffusion. As a result, spatio-temporal edges in these regions are progressively smoothed. Looking at one frame we can observe that the intensity of the edges becomes close to their entourage, but the same happens across frames. Thus, the temporal edges in non-relevant regions are iteratively smoothed and the motion which is not related to the soundtrack is reduced. In fact, by observing the resultant video motion after some iterations we can discover where our algorithm places its attention, that is the possible location of the sound source in the image.

In this section we first define the diffusion ratio α as a measure to quantify the efficiency of the proposed method in removing the video motion that is not related to the sounds in the audio channel. Next, we discuss the value of the parameter K that better suits our objective of keeping only the video information that is needed in joint audio-visual processing.

Let $\mathcal{L}$ be a subset of the video domain Ω: $\mathcal{L} \subset \Omega$. Then, the *amount of motion* M in the video subset $\mathcal{L}$ at iteration n is defined as

$$M^n_{\mathcal{L}} := \sum_{\{i,j,k\}\in\mathcal{L}} |\delta^*_t v^n_{i,j,k}|\,, \tag{4.22}$$

where $|\delta^*_t v^n_{i,j,k}|$ is the absolute value of the temporal derivative approximation $\delta^*_t v$ defined in equation (4.14) at pixel coordinates $\{i,j,k\}$.

We define an audio-visual region of interest (ROI) as the subset of pixels in the video domain that are related to the soundtrack and the complementary region ($\overline{\text{ROI}}$) as the rest of pixels in the video domain: $\text{ROI} \cup \overline{\text{ROI}} = \Omega$. Then, we can define the *audio-visual diffusion ratio* α at iteration n as

$$\alpha^n = \left[\frac{\frac{M^0_{\overline{\text{ROI}}}}{M^n_{\overline{\text{ROI}}}}}{\frac{M^0_{\text{ROI}}}{M^n_{\text{ROI}}}} \right]_{a^{ON}}, \tag{4.23}$$

where the value $M^0_{\text{ROI}}/M^n_{\text{ROI}}$ is the ratio between the amount of motion *inside* the region of interest at iterations 0 (original motion) and n, and $M^0_{\overline{\text{ROI}}}/M^n_{\overline{\text{ROI}}}$ is the same ratio computed *outside* this region of interest. Finally, $[\cdot]_{a^{ON}}$ indicates that only the frames where the audio channel is active (a^{ON}) are used in the computation of this ratio. The audio channel is considered active when sounds are captured by the microphone and thus the normalized audio feature is large enough: $a(t) > 0.1$ with $a(t) \in [0,1]$. To summarize, the *audio-visual diffusion ratio* α is a relative measure that assesses the ability to attenuate the motion in parts of the video signal that are not related to the soundtrack by comparing it to the diffusion experienced in the audio-visual region of interest, *when sounds are present in the audio channel*. Thus, the ratio α quantifies our efficiency *only* in time slots where sounds are present. $\alpha > 1$ when our method favors regions associated to the soundtrack, $\alpha = 1$ when the video motion is equally eliminated inside and outside the ROI, and $\alpha < 1$ if the diffusion affects more our ROI than the rest of the video signal in non-silent periods. Please notice that obtaining $\alpha > 1$ is an extremely challenging task, especially in sequences where the audio-related motion is less intense than the distracting motion.

Let us now study the diffusion parameter K according to the quantitative efficiency measure α. A normalized audio-video synchrony $s_\sigma \in [0,1]$ is used for this analysis. Figure 4.3 shows the typical evolution through iterations of the diffusion ratio α when the audio-related video motion is similar in magnitude to the distracting motion [left] and its evolution when the distracting motion is much more intense and/or spread [right]. Each curve is obtained with a different value of the parameter K. The curves in this figure correspond to sequences in Figures 4.4 and 4.1 respectively,

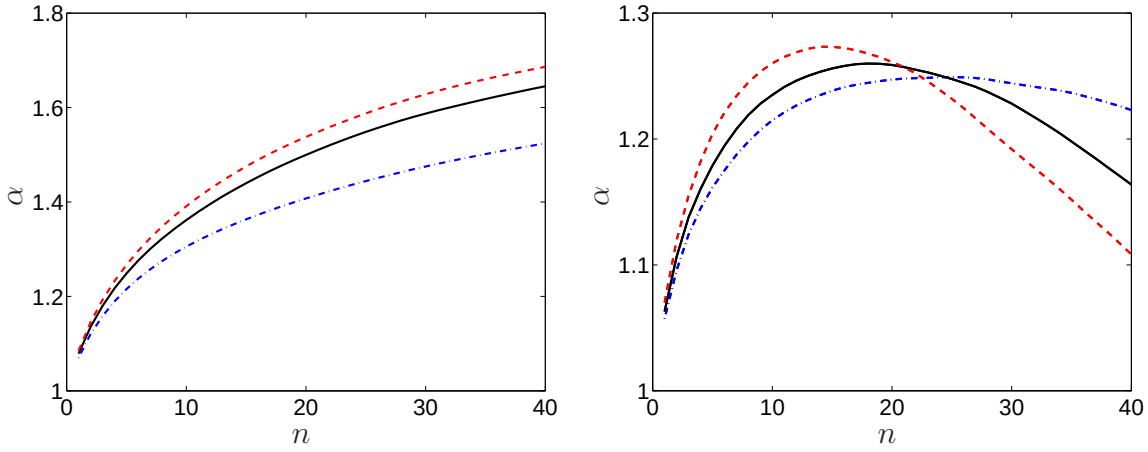
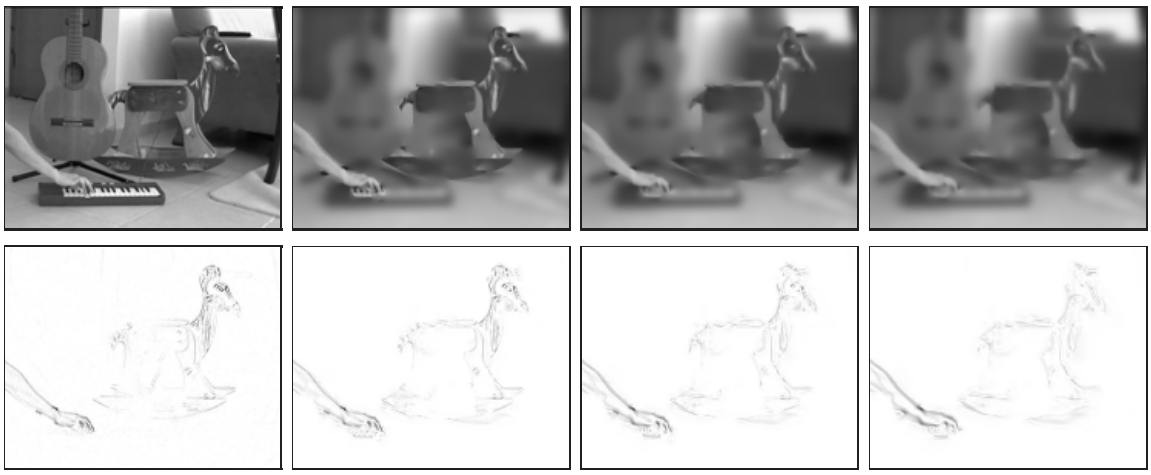


Figure 4.3 – *Evolution through iterations of the* audio-visual diffusion ratio α *when the motion related to the soundtrack has a similar [left] and a much smaller [right] magnitude than the distracting motion. The blue dash dot, black solid and red dashed curves depicted correspond to* $K = 0.05, 0.1, 0.15$ *respectively.*



(a) Original (b) Result $K = 0.05$ (c) Result $K = 0.1$ (d) Result $K = 0.15$

Figure 4.4 – *Results after applying* 30 *iterations of the proposed audio-visual diffusion procedure to a video sequence in terms of pixels intensity [top row] and variation or motion [bottom row] for different values of* K*. In this sequence a hand is playing a synthesizer while a rocking horse generates distracting motion.*

which are taken from the state-of-the-art source localization work presented by Kidron et al. in [36]. In both cases a strong distracting motion is introduced by means of a rocking horse. Thus, while in the first case the magnitude of the audio-related video motion generated by a hand (ROI) playing a synthesizer is similar to the distracting motion, in the second sequence the movements in the mouth region (ROI) are clearly less visible than the rocking horse's ones. As expected α is always above 1 and it reaches a larger value when the distracting motion and the audio-related video motion have similar intensity, i.e. α reaches a value around 1.7 in the left plot (less challenging case) while only 1.27 in the right one. When $K = 0.05$ (small value) the diffusion process evolves slowly in moving regions because there is a lot of irrelevant motion that is taken into account. Our method considers these very small 3D motion concentrations as possibly audio-related and thus it takes time to eliminate them and, as a result, α increases slowly (see the blue dash dot line in Figure 4.3 [left]).

The opposite occurs when K is large ($K = 0.15$). In this case, the diffusion affects most moving regions almost from the beginning and the audio-related motion can be eliminated if it is much smaller than the distracting motion. An example is shown in Figure 4.3 [right] (red dashed line), where α increases fast but then it decreases fast also. A good compromise can be obtained by fixing $K = 0.1$. In this case, the video volume evolves quite fast in moving regions, a high audio-visual diffusion ratio α is reached faster when the audio-related video motion has a similar magnitude than the distracting motion (see the black solid line in Figure 4.3 [left]), and the results when the distracting motion is dominant are also good enough. Remember that Figure 4.3 [right] corresponds to a very difficult case where the audio-related video motion is hardly visible. To sum up, when we fix $K = 0.1$ our approach provides a good efficiency in removing distracting motion while requiring a small number of iterations (low computational cost).

Let us now visually study the effect of varying the parameter K in the evolving video volume and the resulting motion. Figure 4.4 shows the result when applying 30 iterations of the proposed audio-visual diffusion procedure to a sequence where the audio-related video motion and the distracting motion have similar magnitude (bottom left picture). First of all, remember that after the diffusion process the initial frame is blurred and the edges that remain sharp indicate the possible regions of interest for audio-visual analysis that our algorithm identifies. In the resulting frames, the information in the background and static regions is efficiently removed, and the rocking horse is more or less blurred depending on K. In contrast, our region of interest, i.e. the hand playing a synthesizer, is clearly defined in all figures. More details are present in the diffused video volume when using lower values for K (see Figure 4.4 [top]). Indeed, after 30 iterations most of the rocking horse is still salient for $K = 0.05$ while it is blurred when $K = 0.15$. In all cases the motion generated by the hand playing a synthesizer is better preserved than the distracting motion. In the resulting motion pictures the rocking horse's silhouette is less visible when $K = 0.15$ since higher values for K lead to faster diffusion of moving regions. Finally, Figure 4.4 shows that $K = 0.1$ seems appropriate both in terms of pixels' intensity and in motion: the hand silhouette is clearly visible, with sharp edges and a high resolution, and the rocking horse is mostly removed. Furthermore, this value is also appropriate when the distracting motion has a higher magnitude than the audio-related motion as shown in Figure 4.3 [right]. As explained before, a larger value for K can lead to the blurring of audio-related video regions in this challenging case.

4.6 Stopping Criterion

Figure 4.3 shows the necessity of defining a stopping criterion for our iterative method in order to obtain a good audio-visual diffusion ratio α while avoiding to spend an excessive computational time. Indeed, the diffusion process keeps on removing information in the video signal through iterations, and if it was not stopped it would finally blur the whole signal, removing also the audio-related parts and leading to a bad ratio α. In this section we define a stopping criterion for the audio-visual diffusion process which is intuitive and has a low computational cost. This stopping criterion is related to evolution through iterations of the *amount of motion* in the video domain ($M^n_\Omega := M^n$) defined in equation (4.22). Notice that this value is decreasing since at each point the absolute value of the discrete temporal derivative $|\delta^*_t v|$ is bounded by

$$|\delta^*_t v^n_{i,j,k}| \leq \frac{(v_M)^n_{i,j,k} - (v_m)^n_{i,j,k}}{2\Delta t}\,, \tag{4.24}$$

which is monotonically decreasing (see the maximum and minimum result in equation (4.19)). As explained before, our method iteratively eliminates the motion in regions that are not related to the

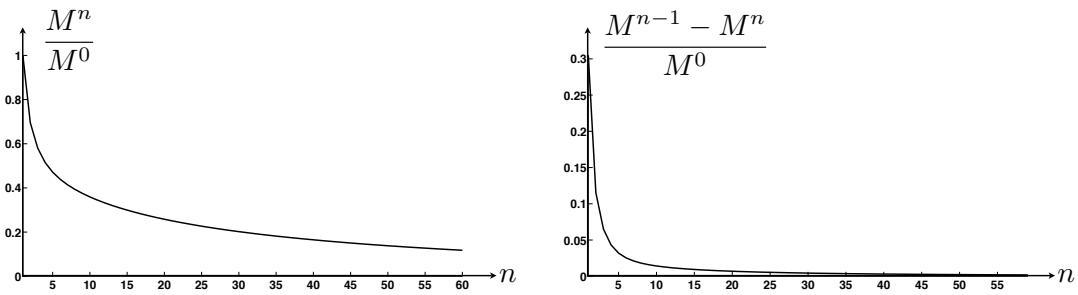


Figure 4.5 – *Typical form of the evolution through iterations of the amount of motion in the video signal [left] and the corresponding motion reduction [right].*

audio signal, leading thus to a global reduction of the motion in the video domain. This behavior can be observed in the graphic shown in Figure 4.5 [left]. In this case for example, only the 20% of the original amount of motion M^0 is kept after $n = 40$ iterations. The rest (80%) is considered as non-related to the soundtrack and it is iteratively removed. The shape of the curve in Figure 4.5 [left] depends on the parameters choice. Thus, for example a higher $\Delta\tau$ represents a faster decrease in M^n since we converge faster towards the solution. In any case, the decrease on the amount of motion is smaller through iterations, tending towards a more or less stable value.

According to this observation, we define the *motion reduction* ΔM at iteration n as

$$\Delta M^n := \frac{M^{n-1} - M^n}{M^0} \,. \tag{4.25}$$

This relative value denotes the percentage of the video motion that is eliminated by our algorithm at iteration n. Thus, when the amount of motion does not decrease significantly $\Delta M^{n_{stop}} < \epsilon$ we stop the diffusion process since we consider that most of the information in regions that are not related to soundtrack has already been eliminated and we are close to the resultant motion map. Figure 4.5 [right] represents a typical shape of the evolution of ΔM through iterations. The 30% of the video motion has been removed at the end of iteration 1, while iteration 10 only eliminates the 1% of the original motion. The constant ϵ has been fixed to $\epsilon = 0.005$. Here we consider that a reduction of 0.5% is not worth the computation of another iteration since it does not change the motion map in a significant way.

The visual effect of stopping the diffusion process before and after the stopping time n_{stop} is shown in Figure 4.6. It depicts the evolution of a video frame and the corresponding motion before and after applying $10, 26$ and 60 iterations of our method ($K = 0.1$ as suggested in Section 4.5). In this sequence, the audio-related video motion in the speaker's mouth is almost impossible to distinguish in Figure 4.6 [bottom left], while the rocking horse's distracting motion is clearly visible. Even in this difficult context, the proposed method is able to handle the situation: the magnitude of the mouth motion grows in comparison to the distracting motion until achieving similar values after 26 iterations. Thus, if the diffusion process is stopped too early (after 10 iterations) the distracting motion is still dominant. In contrast, increasing the number of iterations from 26 to 60 does not change significantly the motion distribution but it duplicates the computational time. Regarding the pixels intensity, the horse features that are still clear after 10 iterations are much more difficult to appreciate when the audio-visual diffusion process advances. Furthermore, the curve depicted in Figure 4.3 [right] shows that the audio-visual diffusion ratio α decreases when increasing the number of iterations for $n > n_{stop}$. As a result, there is no need to keep on diffusing the video volume since after this point the efficiency in removing distracting motion is lower (audio-related regions start to

Figure 4.6 – *Effect of applying the proposed diffusion procedure ($K = 0.1$) an increasing number of iterations ($n = 0, 10, 26, 60$ from left to right) in terms of pixels intensity [top] and motion [bottom]. In the depicted frame a person is speaking while a rocking horse generates distracting motion. The third column corresponds to the signals obtained when applying the proposed stopping criterion ($n_{stop} = 26$).*

be eroded too).

4.7 Evaluation

In this section, we evaluate the efficiency of the proposed nonlinear diffusion approach in favoring the relevant information for joint audio-visual processing. Our objective is to keep the information in regions whose motion is related to the sounds in the audio channel while eliminating details that are not needed. A set of experiments has been performed in audio-visual sequences of different nature presenting strong visual distractors. All the sequences are composed of two moving objects, and only one of them is related to the soundtrack. Our objective is thus to highlight the region in the audio-related object whose motion generates the recorded sounds. In the analyzed sequences the distracting motion is either periodic or with similar characteristics than the audio-related motion.

MovieA and MovieB are the audio-visual sequences that were used respectively in Sections 4.5 and 4.6 to discuss the effect of the diffusion parameter K on our method's efficiency and the need of fixing an automatic stopping criterion for the diffusion process. They are taken from the state-of-the-art source localization work presented by Kidron et al. in [36]. Both sequences are composed of a moving object associated to the audio signal and another one that represents a strong periodic visual distraction (a rocking wooden horse). In MovieA the audio signal is generated by a hand playing a guitar and then a synthesizer, while in MovieB we can see a person speaking and the audio signal is corrupted by the voice of another person. Both video sequences are sampled at 25 frames/sec at resolution of 576×720 pixels and the audio at 44.1 kHz. For its analysis, the video signal has been resized to 144×180 pixels. Each sequence is 10 seconds long approximately.

MovieC is a synthetic sequence composed of a fragment of clips g01 and g08 from the *groups* partition of CUAVE database [62]. This sequence depicts two persons in front of a camera: one of them is uttering numbers while the other one is mouthing the same numbers. The audio signal in this sequence corresponds to the left person in Figure 4.7 [bottom left]. Thus, in the scene we have again one object (person) contributing to the soundtrack and one strong audio-visual distractor.

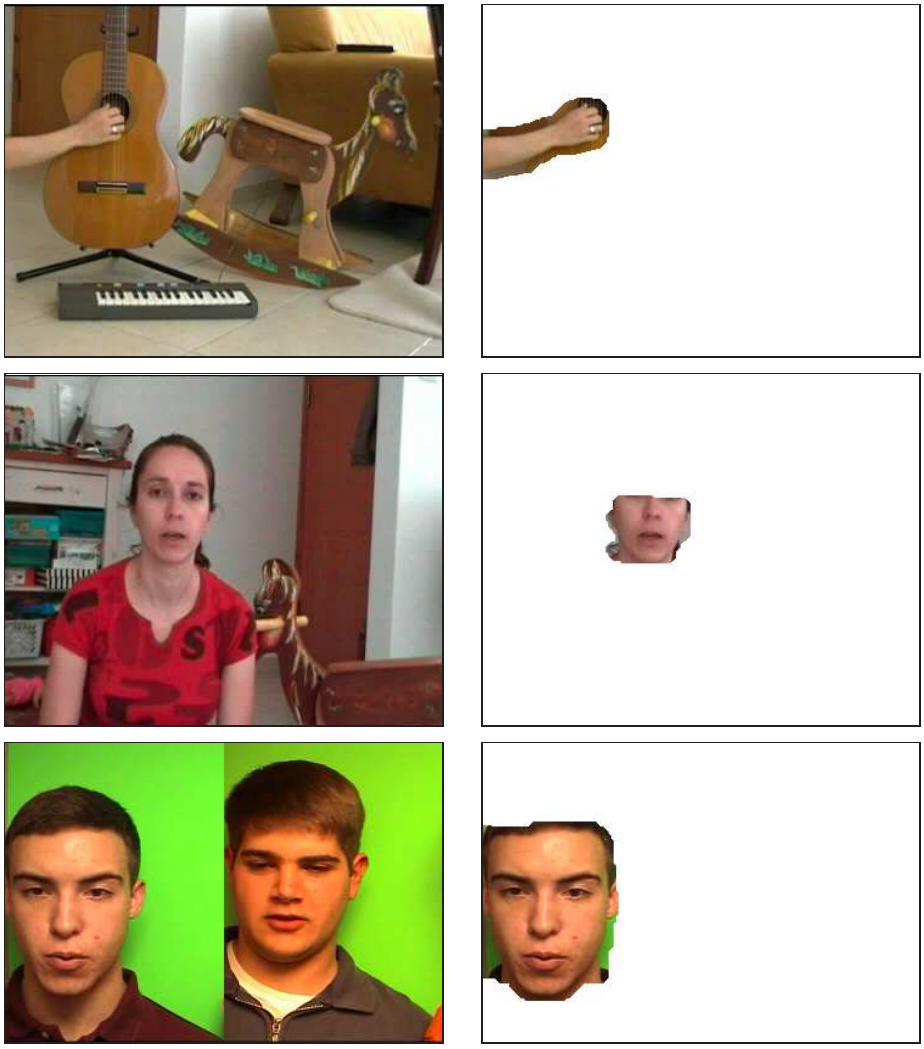


Figure 4.7 – *From top to bottom: frames belonging to MovieA, MovieB and MovieC [left] and corresponding manually defined regions of interest (*ROI*) [right] used to evaluate quantitatively the proposed method. White regions in the right column depict parts of the image not related to the soundtrack (*$\overline{\text{ROI}}$*).*

However, in this case the motion generated by the distractor and the audio-related object have very similar characteristics. The video part of MovieC is sampled at 29.97 frames/sec with a resolution of 480×720 pixels, while the audio part is sampled at 44 kHz. For its analysis, the video signal has been resized to 120×176 pixels. This sequence is around 6 seconds long.

This section is organized as follows. Subsection 4.7.1 presents a *quantitative* evaluation of the performance of our method under all these challenging conditions. In a first stage the regions of interest (ROIs) where the motion is related to the sounds are manually defined. Then, we demonstrate that, as expected, the video motion is better preserved in the ROIs. For this purpose, we use the efficiency measure α that was previously defined in Section 4.5, which computes the ratio between the motion decrease *inside* and *outside* the ROI when sounds are present. Next, in Subsection 4.7.2 we test the robustness of our method when the video signal is visually or acoustically degraded. We show that the proposed audio-visual diffusion process is stable, leading to very similar results in the clean and noisy cases.

We use the same parameters in all experiments. As explained in Section 4.3, we fix $\sigma = 1$ to avoid artifacts due to noise and ensure spatio-temporal coherence. The parameter that controls the diffusion speed has been fixed to $\Delta\tau = 0.15$ since, as discussed in Section 4.4, we need $\Delta\tau \in [0, 1/6]$ to satisfy the maximum and minimum principle in equation (4.19). The *audio-visual synchrony* is normalized $s_\sigma \in [0, 1]$. For the quantitative analysis in Section 4.7.1 different values of K ranging between 0.05 and 0.15 have been used for comparative purposes. However, the rest of experiments in this section have been performed with $K = 0.1$ following to the reasoning in Section 4.5.

Videos showing the test sequences and the corresponding video signals after applying our method are available online at `http://lts2www.epfl.ch/~llagoste/AVdiffusion_results.htm`.

4.7.1 Quantitative Evaluation

Here we provide a quantitative evaluation of the proposed method's efficiency in the complex cases that have been presented before. This quantitative evaluation is performed using the *audio-visual diffusion ratio* α defined in Section 4.5, which compares the amount of video motion removed *in* an *out* of some region of interest (ROI). As explained before, the regions of interest for joint audio-visual processing are defined in this work as spatio-temporal regions in the video signal whose motion generates the sounds captured with the microphone. Figure 4.7 shows a frame belonging to each test sequence and the corresponding ROI in this frame. From top to bottom, the ROI in MovieA corresponds to the hand that plays the guitar and the piano, in MovieB it is defined as the speaker's mouth region, and it is the speaker's face in MovieC. The depicted ROIs have been manually defined using a 3D video segmentation interface.

Results obtained when analyzing MovieA, MovieB and MovieC with the proposed method are shown in Figure 4.8. The original frames of those sequences in (a) present a lot of irrelevant background details such as a carpet or small objects in the shelves that completely disappear or become blobs in the resulting frames in (b). Even if the rocking horse is moving continuously, its silhouette is blurred and most of its details disappear equally. In contrast, the hand that is playing the music instruments in MovieA is well-defined and its limits remain sharp in all cases, even though its motion when playing the guitar is difficult to appreciate. The same happens in MovieB, where the focus in the girl's mouth is preserved despite of its small motion. In MovieC the correct speaker's mouth (left person in the image) is highlighted during all the sequence while the other person's face is mostly blurred. Only when both persons move their mouth exactly at the same time the focus appears also in the distracting person's mouth. However, we cannot consider this as an error because both persons could have been uttered this word, i.e. both movements are coherent with the sound. Let us now discuss the video motion corresponding to the original and resulting signals on those frames, which are depicted in (c) and (d) respectively. In all cases the motion is better preserved in the audio-related video regions, even though some situations are really challenging because the distracting motion is much larger. One consideration should be done concerning MovieB. As explained before, the audio signal in this sequence is corrupted by a second voice. However, the audio feature is not affected by the person speaking out of the field of view, since the energy of this second voice is significantly smaller than the energy of the girl's voice. As a result, this distracting audio signal does not affect significantly the result and the video signal is focused on the girl's mouth only when she is speaking.

Table 4.1 shows the audio-visual diffusion ratio α corresponding to the three analyzed audio-visual sequences when using different values for the parameter K in equation (4.8). Let us recall that $\alpha > 1$ if the video motion is kept more efficiently inside the ROIs when the audio channel is active. As expected, in all experiments we obtain satisfactory values for the quantitative measure

(a) Original frame (b) Resulting frame (c) Original motion (d) Resulting motion

Figure 4.8 – *Results obtained when applying our method to MovieA, MovieB and MovieC with* $K = 0.1$. *The diffusion process has been automatically stopped after* $n_{stop} = 29, 26, 14$ *iterations respectively according to the stopping criterion in Section 4.6.*

	MovieA	MovieB	MovieC
$K = 0.05$	1.41 (20)	1.25 (24)	1.50 (9)
$K = 0.1$	1.58 (29)	1.24 (26)	1.52 (14)
$K = 0.15$	1.64 (32)	1.22 (27)	1.38 (21)

Table 4.1 – *Obtained audio-visual diffusion ratio* α *for the three analyzed sequences when using different values for the parameter* K. *The number of iterations that are required according to the stopping criterion are shown in parenthesis.*

($\alpha > 1$). As discussed before, $K = 0.15$ leads to the best ratio α in MovieA because the distracting noise, with a similar magnitude than the audio-related motion, is removed faster. In contrast, in MovieB the best result is achieved using $K = 0.05$, since in this case the lowest value of K preserves better the motion blobs with low intensity, such as the motion generated by the speaker's mouth. Finally, $K = 0.1$ leads to a good performance in all situations and the best result in MovieC. The

values in parenthesis in Table 4.1 indicate the number of iterations that are required according to the stopping criterion in Section 4.6. In all cases the lowest values correspond to the lowest K ($K = 0.05$) since the amount of motion in the video signal decreases slowly. In this case, a lot of motion blobs are considered as possibly related to the soundtrack, it takes time to discard them and the motion in the video volume evolves so slowly that after some iterations the motion map seems already stuck. A very high number of iterations would be required in order to reach the same level of diffusion in the edges delimiting the moving objects.

These experiments illustrate also the limitations of our approach. In fact, when the analyzed sequence contains a distracting motion which is very consistent with the soundtrack, our algorithm is not able to remove it. An example can be found when the two speakers in MovieC utter a word exactly at the same time. In this case, the focus is kept in the mouths of both speakers because they could *both* be the sound source. In fact, we could be hearing two words, one uttered by each speaker, and the audio feature would not change. The only information that might help in discarding one of them is the knowledge about the frequency characteristics of their voices. However, here we want to keep our method general. Our goal is to focus on the possible sources by eliminating non-relevant information given the assumption of synchrony between audio and video channels. No additional assumptions are used in this approach.

4.7.2 Sequence Degradation

MovieC is used in this section to demonstrate the robustness of our approach under severe acoustic and visual noise conditions. Let us first define the noise measures used for audio and video signals.

The amount of noise in the video signal is expressed in terms of *peak signal-to-noise ratio* (PSNR). Let $\hat{z}$ be a noisy approximation of the discrete 3D video signal z, then the PSNR in dB is computed as

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{(\max_{i,j,k}\{(z, \hat{z})_{i,j,k}\})^2}{\mathrm{MSE}} \right) , \tag{4.26}$$

where MSE is the *mean squared error* defined as

$$\mathrm{MSE} = \frac{1}{N} \sum_{ijk} |z_{i,j,k} - \hat{z}_{i,j,k}|^2 . \tag{4.27}$$

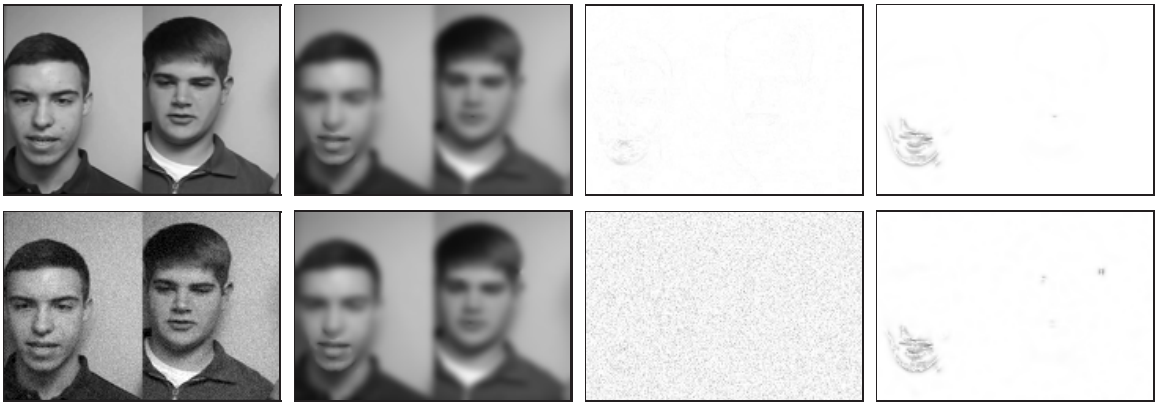
Here N is the total number of pixels in the 3D signal. The more similar are the signals z and $\hat{z}$, the higher is the PSNR.

Then, the amount of noise in the audio signal is expressed in terms of *signal-to-noise ratio* (SNR), which is defined as

$$\mathrm{SNR} = 10 \log_{10} \left(\frac{P_{signal}}{P_{noise}} \right) , \tag{4.28}$$

where P_{signal} and P_{noise} denote the average power of the clean signal and the noise respectively.

Figure 4.9(a) shows a clean frame of this sequence [top], and the same frame when the video signal is corrupted with a white Gaussian noise representing a PSNR = 30 dB [bottom]. In the depicted frame, the video noise is so strong that the real video motion is very difficult to distinguish between the noisy motion (see Figure 4.9(c)[bottom]). The resulting signals after applying 15 iterations of the proposed audio-visual diffusion method to the clean [top row] and noisy [bottom row] video signals are shown in Figures 4.9(b),(d). As expected, even in these challenging conditions our method is able to converge towards a very similar result, where the mouth of the speaker (left person in the image) is highlighted. In effect, the Figures 4.9(b) and (d) are extremely close. The main difference between the resulting signals in this frame is concentrated around the silent person's



(a) Original frames PSNR = 30 dB | (b) Resulting frames PSNR = 47 dB | (c) Original motion PSNR = 23 dB | (d) Resulting motion PSNR = 40 dB

Figure 4.9 – *Effect of adding visual Gaussian noise to MovieC in terms of pixels' intensity and motion. Top row shows a frame of the original and resulting signals when no noise is present, while bottom row depicts the same frame in the noisy case. The number of iterations is* 15 *in both cases. The global* PSNR *comparing the 3D video signals in the top and bottom row is shown below.*

head (right side in the figures), where some noisy motion is interpreted by our method as real motion coherent with the sounds. Furthermore, the resulting motion around the speaker's mouth seems to be a little bit more noisy when analyzing the corrupted signal (see Figure 4.9(d) [bottom]). In both cases the values for the PSNR when comparing the resulting 3D signals are very high. We obtain a PSNR = 47 dB and a PSNR = 40 dB when comparing the diffused video and the resulting motion respectively. This represents an improvement of 17 dB both in terms of pixels' intensity and motion, even though the improvement when visually comparing the original and resulting motion seems much more impressive.

A similar test has been performed in order to evaluate the robustness of our method to acoustic noise. In this case, the audio signal is corrupted with a white Gaussian noise representing a SNR = 5 dB. Figure 4.10 depicts the soundtrack and corresponding audio feature in the clean [top row] and noisy [bottom row] cases. Notice that even though the soundtrack is severely corrupted the proposed audio feature (*equalized* audio energy) does not change much. Thus, after the audio-visual diffusion procedure the resulting video signals are also extremely close (images are omitted in this case since no difference can be observed). The values for the PSNR when comparing the resulting signals in the clean and noisy case are very high. The PSNR = 58 dB when comparing the resulting pixels' intensity and PSNR = 49 dB for the resulting motion.

4.8 Discussion

In this chapter we have introduced a novel algorithm that implicitly combines the information in audio and video channels through PDE-based diffusion. Our method is able to automatically highlight parts of a video signal that are related to a synchronously recorded soundtrack while removing information which is not useful for joint audio-visual processing. The video sequences are simplified using a nonlinear diffusion procedure that integrates the main knowledge in the audio-visual domain: related events in audio and video channels occur approximately at the same time. The proposed diffusion process is controlled by a diffusion coefficient which depends on an estimate

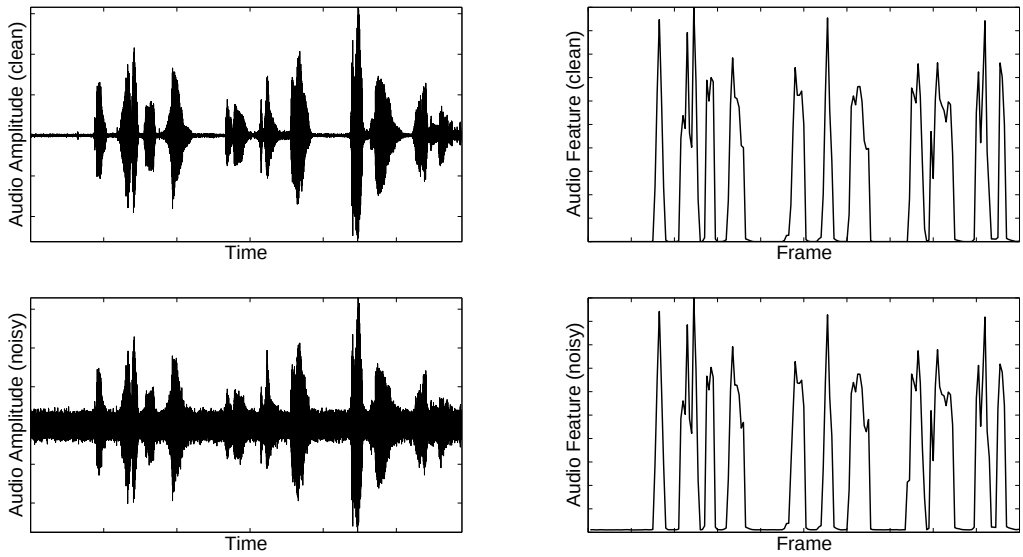


Figure 4.10 – *Soundtrack belonging to MovieC [left] and extracted audio feature* $a(t)$ *[right] before [top] and after [bottom] corrupting the signal with a Gaussian noise (*$\mathrm{SNR} = 5$ *dB).*

of the synchrony between video motion and audio energy at each point of the video domain. Thus, information in video regions presenting low coherence with the soundtrack is iteratively removed. We have introduced a discretization scheme that ensures the numerical stability of this diffusion procedure. A measure of the effectivity of our approach in favoring information in audio-related video regions has been proposed and later used to discuss appropriate values for our method's main parameter. Finally, an intuitive stopping criterion based on the video motion reduction has been introduced in order to automatically stop the diffusion process at an appropriate time.

Several tests have been performed in sequences of different nature presenting real challenges such as moving objects that are not related to the soundtrack and signal degradation. After a few iterations of the proposed method, the sound sources are naturally highlighted. Quantitative results show that our approach is effective in prevailing video regions related to the soundtrack over other moving objects. However, the proposed method is unable to distinguish between two regions whose motion is coherent with a sound. When two persons mouth a word at the same time for example, both mouth regions are highlighted independently of which voice we hear. This audio-visual diffusion procedure is only based on the assumption of synchrony between audio and video channels. No other knowledge about the sources is used because we want to keep this method as general as possible, allowing its application to complex problems in joint audio-visual analysis. We believe that this approach can be efficiently used as a preprocessing step for most methods in this domain, since it is able to remove information that could lead to errors in applications such as sound source localization.

Application: Unsupervised Extraction of Audio-Visual Objects

5

5.1 Introduction

In Chapter 4 we have introduced a nonlinear diffusion approach that naturally highlights video structures correlated to the soundtrack. This audio-visual diffusion procedure leads to a reduction of the video motion in regions presenting a low coherence with the soundtrack. Thus, the motion in the resulting signal represents a good indicator of the possible locations of the sound sources in the image. The objective of this chapter is to use this information in order to extract, in an unsupervised way, the audio-visual objects that are captured by the camera. In other words, we want to extract the video part of the audio-visual sources present in the analyzed scene.

Let us consider the case of speakers for example. Video cameras in laptops are commonly used for video chatting through internet, they can be used in cellphones for mobile video calls and the industry of video game consoles might want to integrate the user face in the games at some point. In this case, our solution would allow the extraction of a video signal containing *only* the speaker's face. This compact video signal contains all the elementary information but the cost of sending it is lower (which might be interesting in the cell phone case) and it can be easily introduced in (augmented reality) video games and even online games using more realistic avatars.

Here we propose first to determine possible regions of interest by comparing the video motion after the audio-visual diffusion procedure to the original motion and then use this knowledge as a starting point for an *audio-visual* segmentation procedure using graph cuts. The extracted region would thus contain the video parts whose motion is highly synchronous to the soundtrack that are identified by the proposed method. Significant progress has been made in the last 20 years in the user-guided foreground/background segmentation domain. Among all segmentation techniques (snakes, active contours, geodesic active contours, shortest path techniques...) graph cuts have shown applicability to N-dimensional problems and flexibility in the definition of the energy to minimize. Furthermore, they provide a globally optimal segmentation through a numerically robust minimization procedure. Graph cuts were first introduced by Boykov and Jolly in [12] for

A preliminary work on this subject can be found in [48].

monochrome N-D signals and extended to color images and videos in latter approaches [4, 39, 68]. For a detailed introduction to the recent advances in image and video segmentation, please refer to the report in [83]. However, all these methods share a limitation: they need the user to provide a starting point for the segmentation process.

In our case, the fusion between audio and video modalities described in Chapter 4 provides this prior information in an unsupervised way. In a first stage, regions presenting a high coherence with the audio channel are automatically classified into the audio-visual object. Then, the remaining pixels are binary classified into the object or the background by means of a novel audio-visual segmentation procedure that keeps together pixels in regions moving synchronously with the soundtrack. The proposed 3D graph cut segmentation is applied within groups of frames (GoF), ensuring thus the spatio-temporal consistency of the extracted region. We propose a sequential application of this procedure in real (long) sequences: the knowledge obtained through the segmentation of a GoF is integrated into the processing of the following one by means of a computationally inexpensive procedure. The extracted region in a GoF is thus influenced by the previous GoF segmentation result *and* the information obtained from joint audio-visual processing.

After the preliminary work of Hershey and Movellan in [33], numerous approaches performed a joint analysis of information in audio and video modalities in order to locate the sound sources in the image [26, 36, 58, 74–76]. In contrast, only the method that we presented previously in Chapter 3, and the works of Liu and Sato in [40, 41] attempted the extraction of the source's video part. In Chapter 3 the video signal is decomposed into basic image structures (atoms), next the sources position is estimated by clustering together atoms with high audio-visual correlation, and finally the video part of each source is reconstructed by adding the contribution of the atoms that are close to its estimated position. Thus, in Chapter 3 the particular shapes of the sources are not considered, i.e. the extracted audio-visual objects have always an approximately circular shape because all atoms inside a radius are used in the source's reconstruction process. In [40, 41] Liu and Sato overcome this limitation by using a segmentation technique based on graph cuts, which is initialized by joint audio-visual analysis. In their first work [40] the source position is estimated by computing the quadratic mutual information between audio and video features, and this procedure is applied to sequences composed of almost static speakers. Then, in [41] this method is generalized to non-stationary sources by identifying the pixel's visual trajectories whose changes in acceleration better fit the energy variations in the audio channel.

Let us now explain in detail the main contributions of our approach.

1. From a video segmentation point of view, the introduction of *audio-visual priors* makes the segmentation automatic. As explained before, the need for user interaction is the main limitation of previous segmentation approaches [4, 12, 39, 68].

2. We propose an innovative *audio-visual term* in the energy function that the graph cut algorithm minimizes. The term presented in [40, 41] forces the regions presenting low correlation with the soundtrack to be part of the background, by promoting global links rather than links between neighboring pixels. In contrast, our audio-visual term does not affect regions with low coherence and it does not include any implicit assumption about these regions. As a result, in our case the audio-visual object can be completely extracted even though some parts of it present a lower audio-visual coherence. This point is further discussed in Section 5.3.

3. We redefine the standard *regional term* in the segmentation energy function, which integrates knowledge about the color distributions in foreground and background. In Section 5.3 we demonstrate the advantages of the proposed regional term over the commonly adopted term

in [12, 39, 68]. Furthermore, keeping a regional term in the energy function represents an advantage over previous *audio-visual* segmentation approaches in [40, 41], since this term ensures a higher cohesion between the homogeneous regions that typically compose an audio-visual object.

4. Unlike in [40, 41], here we consider the problem of extracting the audio-visual object in an *entire* sequence (not only a fragment of it). For this purpose, the video signal is divided into Groups of Frames (GoFs) which are processed separately but not independently. We propose an implementation that allows computationally inexpensive propagation of the segmentation results through time. The 3D characteristic of the proposed segmentation procedure ensures coherence between neighboring frames and makes unnecessary the utilization of additional shape terms, which are typical in frame by frame approaches as in [4].

This chapter is structured as follows. In Section 5.2 we define the audio-visual coherence, which quantifies the relationship between video structures and sounds at the pixel level. Section 5.3 explains the proposed segmentation of a GoF, which introduces an energy term that integrates the knowledge obtained from joint audio-visual processing. In Section 5.4 we present an automatic criterion to choose the segmentation priors according to the audio-visual coherence. Section 5.5 introduces the methodology that is applied to extract the audio-visual objects from an entire sequence by propagating the segmentation results forward in time. Section 5.6 presents the experiments performed on challenging audio-visual sequences presenting non-stationary sources, distracting moving objects and multiple audio-visual sources alternating their periods of activity. Finally, in Section 5.7 achievements and future research directions are discussed.

5.2 Audio-Visual Coherence

In Chapter 4 we presented a method to selectively remove the information in video regions that are not required for joint audio-visual processing. The 3D characteristic of this nonlinear diffusion procedure eliminates spatio-temporal edges in regions that are not related to the soundtrack and, as a result, the motion in these regions decreases. Then, regions in which the video signal is least diffused can be identified by simply comparing the motion before and after the audio-visual diffusion process. Notice that the regions in which the motion is better preserved are, with high probability, part of the audio-visual object since their movements are correlated to the sounds in the audio channel.

In fact, we could simply use the motion in the resulting video signal as an indicator of the sound sources location. This is a good option if the audio-related video motion and the distracting motion have similar intensities. However, when the distracting motion has a significantly higher magnitude, this motion can still be dominant in the resulting motion map even if our approach preserves more efficiently the audio-related region. Thus, we need a relative value that compares the magnitude of the motion before and after the audio-visual diffusion process.

We define the *audio-visual coherence* $c(\mathbf{x}) \in [0,1]$ at pixel location $\mathbf{x}$ as

$$c(\mathbf{x}) = \begin{cases} \frac{\partial_t v(\mathbf{x},\tau_{stop})}{\partial_t v(\mathbf{x},0)} & \text{if} \quad \partial_t v(\mathbf{x},0) > \xi \\ \frac{\partial_t v(\mathbf{x},\tau_{stop})}{\max_{\mathbf{x}} \partial_t v(\mathbf{x},0)} & \text{otherwise} \end{cases} \tag{5.1}$$

where $\partial_t v(\mathbf{x},\tau_{stop})$ is the temporal derivative of the resulting video signal after n_{stop} iterations of the proposed nonlinear diffusion procedure ($\tau_{stop} = n_{stop}\Delta\tau$), the constant ξ makes the audio-visual

coherence $c(\mathbf{x})$ close to zero in static pixels (we can fix $\xi = 10^{-2}$ for example), and the constant s makes $c(\mathbf{x})$ unitary. Thus, the higher is the *audio-visual coherence* $c(\mathbf{x})$ the higher is the probability for the video pixel at location $\mathbf{x}$ to be part of an audio-visual object, since its motion is well preserved through the diffusion process.

Figure 5.1 depicts the highest values for the original video motion (a), the resulting video motion (b) after the audio-visual diffusion process in Chapter 4 and the proposed audio-visual coherence (c) for the three sequences analyzed in Chapter 4. The movies contain two moving objects, and only one of them is associated to the soundtrack. From top to bottom, the audio-visual objects are the hand playing the synthesizer, the girl and the left boy. Then, the distracting motion is generated by a wooden rocking horse in the first two sequences and by the right person, who is uttering the same numbers than the real speaker in the bottom movie. In the depicted frames either the original video motion in the audio-visual object has approximately the same magnitude than the distracting motion (MovieA, MovieC) or it is significantly smaller (MovieB). Thus, the highest values (white regions) for the original video motion (a) are distributed equally between the hand and the rocking horse's head in MovieA, completely concentrated in the horse in MovieB and situated in the face of both persons in MovieC. As expected, the video motion is efficiently kept in the audio-visual objects through the diffusion process and the resulting audio-visual coherence (c) is higher (darker in the pictures) in these regions. In MovieA only a few white pixels appear over the rocking horse, most of them are concentrated in the girl's mouth in MovieB and again a small number of them are situated on the wrong person in MovieC. Notice that the white pixels corresponding to the 0.5% highest values of the audio-visual coherence in (c) are much more concentrated than the highest values of the original motion in (a). As explained before, the diffusion process in Chapter 4 evaluates the synchrony between video *regions* and the soundtrack and thus it is intuitive that the highest values of the audio-visual coherence are found in regions (white blobs) rather than isolated pixels.

The *audio-visual coherence* $c(\mathbf{x})$ represents an efficient measure of the relationship between video regions and the audio signal, with a high spatial resolution. As a result, it is intuitive to use regions with high audio-visual coherence as a starting point for a 3D segmentation procedure whose objective is to extract audio-visual objects in an unsupervised way. The audio-visual coherence is also used in next section to define a novel graph cut segmentation approach that includes the knowledge extracted from joint audio-visual processing.

5.3 Graph Cut Segmentation using Audio-Video Synchrony

Our 3D segmentation approach is based on the procedure proposed by Boykov and Jolly in [12]. Given some initial information about foreground and background locations provided by the user (seeds) their algorithm computes a globally optimal segmentation of monochrome N-dimensional images using graph cuts. In this section, this procedure has been extended to color video signals by integrating joint audio-visual processing. Our main contribution is the introduction of an *audio-visual term* in the definition of the energy function that we minimize through the graph cuts. The audio-visual term that we propose links together neighboring pixels belonging to a region with high audio-visual coherence. Thus, this term ensures that pixels in audio-visual objects are kept together through the segmentation process. Since the connections between pixels are spatio-temporal, we reinforce also the links between neighboring frames in regions where the image structures move coherently with the sounds. Furthermore, the regional term used in [12, 39, 68] has been redefined and the advantages of the proposed term are demonstrated in this section.

Let $\mathbf{z} = (z_1, \ldots, z_p, \ldots, z_P)$ be the set of pixels in the RGB color space that compose a group

(a) Original motion (b) Resulting motion (c) Audio-visual coherence

Figure 5.1 – *White pixels in the bottom row indicate the* 0.5% *highest values corresponding to the features in the top row: original motion* $\partial_t v(\mathbf{x}, 0)$ *(a), resulting motion* $\partial_t v(\mathbf{x}, \tau_{stop})$ *(b), and audio-visual coherence* $c(\mathbf{x})$ *(c). From top to bottom, in the depicted frames a hand is playing a synthesizer (MovieA), a girl is speaking (MovieB) and the left guy is uttering some numbers (MovieC), while some distracting motion is present. In all cases the highest values of the audio-visual coherence are much more concentrated in the audio-related region.*

of frames (GoF). The segmentation process consists on assigning a foreground opacity level $l = (l_1, \ldots, l_P)$ to each pixel p. In general $0 \leq l_p \leq 1$, but here we perform a hard segmentation and thus our labels are binary $l_p \in \{0(\text{background}), 1(\text{foreground})\}$.

The procedure is the following. First, we build a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ corresponding to a 3D GoF following the procedure in [12]. The set of vertices $\mathcal{V}$ is composed of the pixels $p \in \mathcal{P}^j$ in j-th GoF plus two additional nodes: a foreground terminal F and a background terminal B. The set of edges $\mathcal{E}$ is composed by edges connecting neighboring pixels $\{p, q\} \in \mathcal{N}$ (n-links) and edges connecting each pixel p to the foreground and background terminals $\{p, F\}$ and $\{p, B\}$ (t-links). In our graph the neighborhood $\mathcal{N}$ of each pixel is composed of six pixels, four spatial neighbors and two temporal neighbors as in [12].

Then, the graph cut algorithm solves the segmentation problem by minimizing the following energy defined on the graph:

$$\begin{aligned} J(l) &= \lambda_R\, R(l) + V(l) + \lambda_C\, C(l) \\ &= \lambda_R \sum_{p \in \mathcal{P}^j} R_p(l_p) + \sum_{\{p,q\} \in \mathcal{N}} (V_{p,q} + \lambda_C C_{p,q})\, [l_p \neq l_q] \;, \end{aligned} \tag{5.2}$$

where $[\Phi]$ denotes the indicator function taking values 0, 1 for a predicate Φ. The regional term $R(l)$ evaluates how the color z_p corresponding to each pixel p with label l_p fits into the background and foreground models, the boundary term $V(l)$ assesses the similarity of each pixel with its neighborhood, and the audio-visual term $C(l)$ links together neighboring pixels belonging to a region presenting a high audio-visual coherence. The coefficients λ_R and λ_C define the relative importance of the regional term and the audio-visual term with respect to the boundary term. In all experiments this parameters have been fixed to $\lambda_R = 0.05$, a value within the range defined by [39] and [68] (0.07 and 0.02 respectively), and $\lambda_C = 0.6$ so that the audio-visual term is less important than the boundary term ($\lambda_C < 1$) and thus the extracted region respects the strong edges in the image.

The energy $J(l)$ is minimized using the Boost Graph Library implementation [11] of the classical minimum cut algorithm in [12].

As explained in Section 5.1 Liu and Sato introduced an energy term that included the knowledge obtained by fusing audio and video modalities in order to extract the speaker face region [40] or general sound sources [41]. In a first stage, the Expectation Maximization algorithm was used to cluster the audio-visual correlation values into two clusters, the first cluster representing the sound source and the second one the background. Then, they proposed to replace the standard regional term $R(l)$ in equation (5.2) by a cost to assign a pixel to be part of the sound source, which depended on the Mahalanobis distance between the pixel and the estimated mean value of the source's correlation. Here in contrast, we propose to keep the regional term (by redefining the one in [12, 39, 68]) and we introduce a novel audio-visual term. Our term links together neighboring pixels in regions with high audio-visual coherence instead of linking each pixel to the foreground and background terminals. The advantages of our configuration are discussed further in this section.

Here we first introduce the *boundary term*, for which we use the standard definition in [12, 39, 68], next we define the *regional term*, which is slightly different from previous approaches, and finally we present our novel *audio-visual term*.

The *boundary term* is defined by

$$V_{p,q} = \frac{1}{dist(p,q)} \exp\left(-\frac{\|z_p - z_q\|^2}{2\gamma_V^2}\right) , \tag{5.3}$$

where $\gamma_V^2 = \mathbb{E}(\|z_p - z_q\|^2)$ as in [68]. Here $\mathbb{E}(\cdot)$ denotes the expectation operator over the video signal and $dist(\cdot)$ is the Euclidean distance between neighboring pixels.

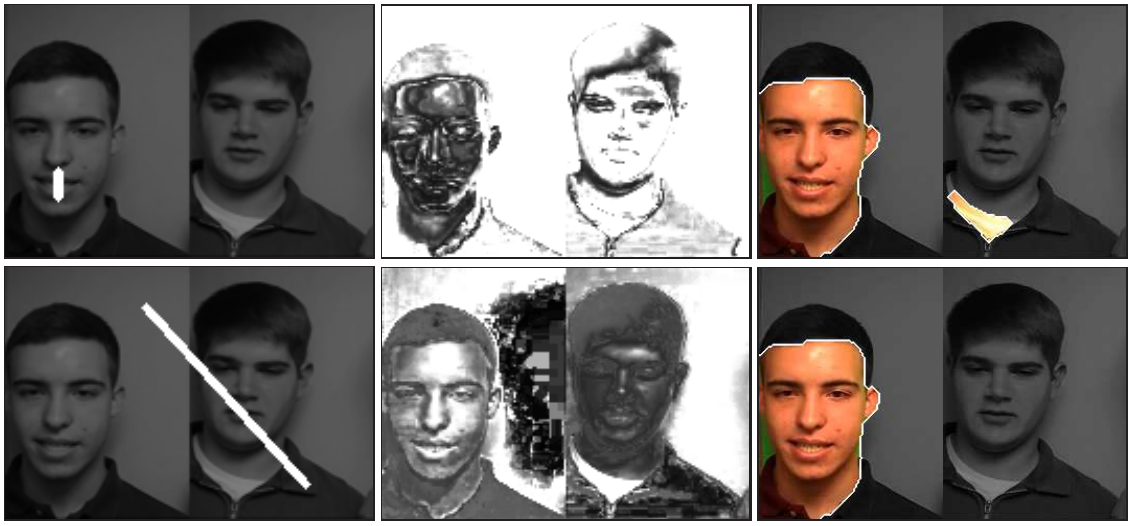


Figure 5.2 – *Segmentation results [right] when using the regional term in previous methods [top] and our regional term [bottom] given the manually-added seeds [left] and the corresponding probability maps [center] for foreground [top] and background [bottom]. The audio-visual term is not taken into account ($\lambda_C = 0$). The segmented foreground is shown in color and the background in a darker grayscale. White regions represent the seeds in the left column and a very low probability in the center.*

Gaussian Mixture Models (GMMs) are used to estimate the foreground (Λ_f^{color}) and background (Λ_b^{color}) color distributions from the available seeds, by using the Expectation Maximization algorithm: $\Lambda_m^{color} = \{u_{m,i}, \mu_{m,i}, \Sigma_{m,i}\}_{i=1}^{Q}$ for $m = \{b, f\}$. For each Gaussian i composing the mixture, u_i, μ_i and Σ_i denote respectively its weight, mean and covariance matrix. The number of Gaussians characterizing the foreground/background models has been fixed to $Q = 5$ in all simulations as in [68]. According to these color models, the penalties for assigning the pixel p to the foreground ($l_p = 1$) and background ($l_p = 0$) that compose the *regional term* have been defined respectively as

$$R_p(l_p = 1) = h(\ln \mathrm{P}(z_p|\Lambda_b^{color})), \tag{5.4}$$

$$R_p(l_p = 0) = h(\ln \mathrm{P}(z_p|\Lambda_f^{color})), \tag{5.5}$$

where $\mathrm{P}(z_p|\Lambda_m^{color})$ is the probability for a pixel p to belong to the foreground/background given the GMM Λ_m^{color} and $h(\cdot)$ is a function that maps $\ln \mathrm{P}(z_p|\Lambda_m^{color})$ from $(-\infty, 0]$ to $[0, 1]$ where "0" and "1" represent the lowest and the highest probability respectively.

Let us now discuss the differences between the proposed regional term and the one introduced in [12] and also used in [39, 68]. In our case, the edge that links any pixel p to the foreground (background) is proportional to the probability that its color z_p belongs to the foreground (background) color model expressed by Λ_f^{color} (Λ_b^{color}). Previous methods used the negative log-likelihoods, and thus the edge linking a pixel p to the foreground (background) was *inversely* proportional to this probability. Figure 5.2 illustrates the advantages of the proposed regional term. The probability for a pixel situated in the right person's shirt of belonging to the foreground and background is very low (in white in the central figures). According to the proposed regional term, the links between those pixels and the background and foreground terminals (F and B) have a very low weight and thus they do not influence the segmentation results. However, when using the term in [12, 39, 68] the link between the pixels in the shirt and the foreground terminal F is much stronger than the link to the background because the probability of belonging to the background is lower. Thus, the weight distribution of previous methods enforced the segmentation algorithm to label those pixels

edge	weight (cost)	for
$\{p,q\}$	$V_{p,q} + \lambda_C C_{p,q}$	$p,q \in \mathcal{N}$
$\{p,F\}$	$\lambda_R \cdot h(\ln \mathrm{P}(z_p \vert \Lambda_f^{color}))$	$p \in \mathcal{P}^j, p \notin \mathcal{F}^j \cup \mathcal{B}^j$
	L	$p \in \mathcal{F}^j$
	0	$p \in \mathcal{B}^j$
$\{p,B\}$	$\lambda_R \cdot h(\ln \mathrm{P}(z_p \vert \Lambda_b^{color}))$	$p \in \mathcal{P}^j, p \notin \mathcal{F}^j \cup \mathcal{B}^j$
	0	$p \in \mathcal{F}^j$
	L	$p \in \mathcal{B}^j$

Table 5.1 – *Proposed weight distribution for the graph.*

as foreground, even though this is not clear at all according the color models. Notice that the segmentation result contains the right person's shirt when applying the regional term in [12, 39, 68], while it is not extracted in our case [bottom right]. In this work, we prefer to rely on the *boundary term* and do not influence the segmentation when the probabilities of belonging to foreground and background are weak.

The proposed *audio-visual term* is defined by

$$C_{p,q} = \frac{1}{dist(p,q)}\, c_p\, \exp\left(-\frac{|c_p - c_q|^2}{2\gamma_C^2}\right), \tag{5.6}$$

where c_p is the audio-visual coherence $c(\mathbf{x})$ corresponding to pixel p with spatio-temporal coordinates $\mathbf{x}$. We fix $\gamma_C = 0.1$ to assign a low weight to links between neighboring pixels with different coherence. Since in this case $C_{p,q} \neq C_{q,p}$ if $c_p \neq c_q$, our graph is directed. The proposed audio-visual term is thus similar to the boundary term in the sense that it is computed between neighboring pixels. Furthermore, low weights are assigned to the edges that link pixels belonging to different regions (in this case regions presenting high and low coherence instead of regions with significantly different color). Our audio-visual term does not affect regions with low audio-visual coherence. Notice that the weight $C_{p,q}$ is directly proportional to the audio-visual coherence in the origin pixel c_p and thus the weight of the links is close to zero in regions with low coherence. As a result, our audio-visual term only links together neighboring points that present a similar and *relevant* audio-visual coherence. This represents the main difference between our *audio-visual term* and the term in [40, 41]. In their case, all the pixels are linked to the background and foreground terminals according to their audio-visual correlation. Thus, when a part of the audio-visual object has a low coherence with the audio signal, the segmentation process assigns this part to the background. However, some applications such as speaker's face extraction might be interested in extracting the speaker's forehand (which is part of the audio-visual object) even though it does not present a high coherence with the speech. Our segmentation approach might be more suitable for these applications since our term links together neighboring regions with high audio-visual coherence without penalizing or making any assumptions about the remaining video regions.

The distribution of the weights in the graph is summarized in Table 5.1. Here $p \in \mathcal{F}^j$ and $p \in \mathcal{B}^j$ denote respectively the set of points in j-th GoF that are classified into foreground and background by the joint audio-visual analysis in Section 5.4 (segmentation seeds). In general, when the seeds are manually fixed we use $L = 1 + \max_{p \in \mathcal{P}^j} \sum_{q:\{p,q\} \in \mathcal{N}} (V_{p,q} + \lambda_C C_{p,q})$ to ensure that the seeds label is not modified as in [12]. However, since in our approach the seeds are chosen in an unsupervised way the weight of the link between the seeds and the corresponding terminal (F or B) is fixed to the

maximum weight of a n-link: $L = \max_{p \in \mathcal{P}^j}(V_{p,q} + \lambda_C C_{p,q})$. This value is high enough to influence the segmentation but the seeds label can be modified by the min-cut max-flow algorithm if required, e.g. when a foreground seed is isolated in the middle of a region labelled as background.

5.4 Estimation of the Audio-Visual Segmentation Priors

The previous section introduced an audio-visual segmentation procedure that integrates information in audio and video channels. This approach requires a starting point for the segmentation process, i.e. some initial information about the foreground (audio-visual object) and background locations. As explained in the introduction, in our approach this prior information is obtained from the fusion of audio and video modalities. In Section 5.2 we presented a measure to quantify, at the pixel level, the synchrony between the motion of image structures and the soundtrack. The higher is the audio-visual coherence $c(\mathbf{x})$ the more probable is that pixel $\mathbf{x}$ belongs to an audio-visual object, since the motion in this region is well preserved through the audio-visual diffusion process. As a result, we can easily identify the pixels (or regions) that are likely to belong to the audio-visual object, as those pixels presenting the highest audio-visual coherence.

Let P be the number of pixels in the video GoF. The number of seeds that are automatically chosen for foreground N_f and background N_b are

$$N_m = PH_m \qquad \text{for } m = \{f, b\}\,, \tag{5.7}$$

where the quantities H_f and H_b can be fixed depending on the application. In our method, the foreground seeds are chosen to be the N_f pixels with highest audio-visual coherence c_p, while the N_b constraints for the background are uniformly distributed at *random* in the GoF. Thus, the choice of the segmentation priors is based on the assumption that regions moving coherently with sounds in the audio channel probably belong to the audio-visual object. The random election of background seeds ensures that no additional assumptions are made. In previous approaches [40, 41] the pixels presenting a low audio-visual correlation were assumed to belong to the background and, in consequence, these regions could not be included in the extracted region.

In all experiments we use $H_f = H_b = 3 \cdot 10^{-3}$, so that a 0.3% of the pixels are automatically labelled as foreground and background. This value is low because we want to be sure to introduce the smallest possible number of errors in the initial labeling. The effect of increasing H_f and H_b on sequences presenting distracting motion is shown in the experiments section. A choice of $H_f > H_b$ can lead to the extraction of larger foreground regions.

In the proposed method, no segmentation seeds are fixed in the video frames in silent periods. These frames are strongly affected by the diffusion process due to the absence of audio energy. As a result, the audio-visual coherence is very low and no foreground seeds are fixed. Thus, the introduction of background seeds in the silent frames would only penalize the extraction of the audio-visual object.

5.5 Audio-Visual Object Extraction on Entire Sequences

The extraction of audio-visual objects in a 3D GoF has been explained in last sections. We first defined the audio-visual coherence between each pixel and the soundtrack, and then we detailed how this information is used in order to choose the segmentation priors and extract the audio-visual object in the GoF. However, real video signals are very big and they can not be analyzed as a

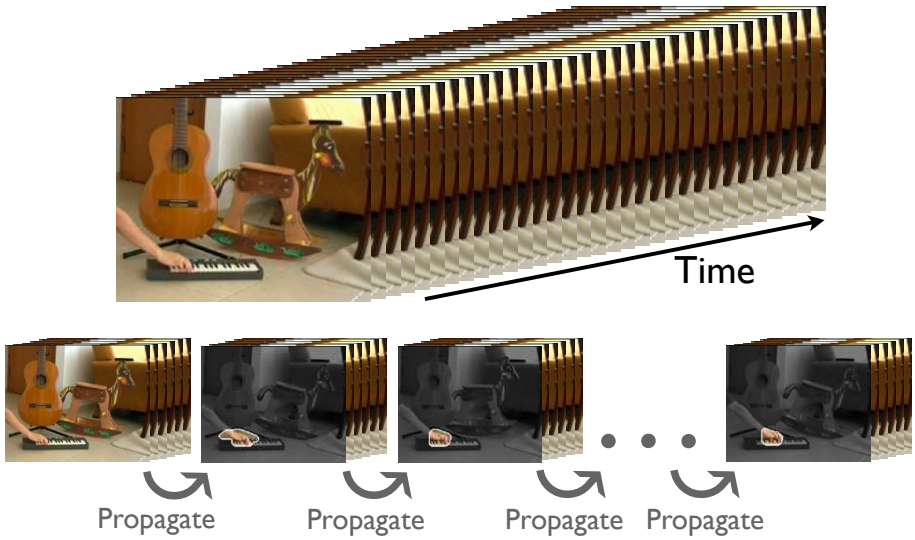


Figure 5.3 – *Scheme illustrating the proposed implementation. The video signal is divided into groups of frames (GoFs), which are processed sequentially. Our method propagates the segmentation from one GoF to the next one by using the segmentation results obtained in the frame that they share.*

whole, i.e. we need to divide the sequences into GoFs and process each GoF separately. In this section we introduce an audio-visual segmentation algorithm that can deal efficiently with long video signals. Our method is based on the propagation of the segmentation results through time. The information that is extracted after the processing of a GoF, such as the location and characteristics of the audio-visual sources in the scene, is used in the following GoF's segmentation. As a result, the GoFs are processed separately *but not independently*. In our configuration, each GoF shares one frame with the previous GoF in order to make easier the propagation of the segmentation results forward in time. The idea is to process the GoFs sequentially: when sounds appear the first GoF is segmented and then the results are propagated forward in time by combining at each step the knowledge extracted from the previous GoF and the joint audio-visual processing on the current GoF. Thus, our approach exploits the temporal coherence between neighboring frames and ensures the continuity of the segmentation results. A scheme illustrating our propagative procedure is shown in Figure 5.3.

The proposed unsupervised audio-visual segmentation algorithm is summarized in Algorithm 2. First, we apply the audio-based nonlinear diffusion process described in Chapter 4, and we compute the audio-visual coherence c_p for each pixel p according to the procedure in Section 5.2. This information represents the starting point for the 3D audio-visual segmentation approach described in Section 5.3 and it will be used in all stages of our algorithm. For its processing, the video signal is divided into fixed-size groups of frames (GoFs), each neighboring pair of GoFs sharing one frame. This configuration is chosen for two main reasons. First, all video GoFs have the same size N_t and thus the same graph structure. As a result, the graph is built for the first GoF and then reused in the next ones (only the weights change). Second and most important, the frame that two neighboring GoFs share allows an easy propagation of the segmentation through time. Indeed, the seeds that are used in the segmentation of a GoF are obtained from the audio-visual analysis in Section 5.4 *and* the segmentation results in the shared frame. Thus, this frame links neighboring GoFs and it allows an intuitive introduction of prior information in the GoF segmentation.

Once these global steps are completed, we compute the segmentation of the first GoF as explained

Input: Video signal $v(\mathbf{x})$ and audio signal $a(t)$
Output: Segmented video: binary labels l

A. Compute the audio-visual coherence c_p for each pixel $p \in \mathcal{P}$ from the audio and video signals $a(t)$ and $v(\mathbf{x})$.
B. Partition the video signal into M groups of frames (GoF). Each neighboring pair of GoFs shares one frame: $\mathcal{P}^j \bigcap \mathcal{P}^{j+1} \neq \emptyset$. The first GoF starts when sounds appear.

for *First GoF ($j = 1$)* **do**
1. Classify the N_f pixels with highest audio-visual coherence into the foreground ($p \in \mathcal{F}^1$) and choose randomly N_b pixels as background seeds ($p \in \mathcal{B}^1$).
2. Learn color models for foreground and background ($\Lambda_{f,1}^{color}$ and $\Lambda_{b,1}^{color}$) from the audio-visual seeds in this GoF.
3. Segment the GoF and obtain the labels l^1 and the corresponding trimap T^1 given the color models $\Lambda_{f,1}^{color}$, $\Lambda_{b,1}^{color}$, and the seeds $p \in \mathcal{F}^1$, $p \in \mathcal{B}^1$.
end
foreach *GoF ($j = 2, \ldots, M$)* **do**
1. N_f and N_b *audio-visual seeds* are fixed for foreground and background ($p \in \mathcal{F}^j$ and $p \in \mathcal{B}^j$ respectively) following the same procedure than for the first GoF.
2. Add N_f^c and N_b^c *continuity seeds* for foreground and background according the segmentation result on the frame that the previous GoF and the current one share as

$$p \in \mathcal{F}^j \quad \leftarrow \quad p \in \mathcal{F}^j \cup \mathrm{R}_{N_f^c}\{\mathcal{C}_f\}\,, \tag{5.8}$$
$$p \in \mathcal{B}^j \quad \leftarrow \quad p \in \mathcal{B}^j \cup \mathrm{R}_{N_b^c}\{\mathcal{C}_b\}\,. \tag{5.9}$$

Here $\mathrm{R}_N\{\psi\}$ denotes the restriction of the set ψ to N of its values chosen uniformly *at random*, and $\mathcal{C}_f$, $\mathcal{C}_b$ are the set of all possible pixels to use as continuity seeds, which are labelled as foreground and background in the shared frame trimap:

$$\mathcal{C}_f \quad = \quad \{\, p \in \{\mathcal{P}^{j-1} \cap \mathcal{P}^j\} : T_p^{j-1} = 1 \,\}\,, \tag{5.10}$$
$$\mathcal{C}_b \quad = \quad \{\, p \in \{\mathcal{P}^{j-1} \cap \mathcal{P}^j\} : T_p^{j-1} = 0 \,\}\,. \tag{5.11}$$

3. Compute color models ($\Lambda_{f,j}^{color}$ and $\Lambda_{b,j}^{color}$) according to the audio-visual seeds in this GoF *and* the continuity seeds in the shared frame.
4. Segment GoF j and obtain the labels l^j and the corresponding trimap T^j given the color models $\Lambda_{f,j}^{color}$, $\Lambda_{b,j}^{color}$ and the seeds $p \in \mathcal{F}^j$, $p \in \mathcal{B}^j$.
end

Algorithm 2: Unsupervised audio-visual segmentation of entire sequences

in the previous sections of this chapter. Notice that the extraction of the audio-visual object starts when the first sounds are captured by the microphone. The seeds are fixed according to the reasoning in Section 5.4: the N_f pixels presenting the highest audio-visual coherence become seeds for the foreground and the same number ($N_b = N_f$) of background seeds are uniformly distributed at random across the GoF. The number of seeds is small in order to introduce a minimum of errors and no seeds are fixed in silent frames. Next, GMMs are estimated for foreground and background color distributions on the available seeds, which are obtained by joint audio-visual processing. Finally, the segmentation is computed according to the procedure explained in Section 5.3. In order to avoid

Figure 5.4 – *Key steps in the processing of an intermediate GoF. Top row illustrates the extraction of the* continuity seeds*: from left to right it depicts the segmentation result in the first frame (obtained from the previous GoF's processing), the corresponding trimap when dilating and eroding the segmentation boundaries, and the foreground and background continuity seeds (white pixels) respectively. Then, the middle row shows the foreground audio-visual seeds obtained for some other frames in the GoF, and the bottom row depicts the extracted regions in these frames. The active speaker was the right person in the previous GoF and it is the left person in the current GoF.*

the propagation of errors through time, the limits of the segmentation are dilated and eroded to build a trimap T indicating locations where the labels have enough confidence. Figure 5.4 [top left] shows a segmented frame and the corresponding trimap, whose value is 1 in the foreground (white), 0 in the background (black) and 0.5 in the border between those two regions (gray).

At this point we could treat each GoF independently by following the same procedure that is applied to the first GoF. However, our objective is to exploit the temporal consistency that characterizes video signals (neighboring frames are usually very similar). We would like to introduce the knowledge extracted from the segmentation in the previous GoF in order to influence (but not determine) the results in the current GoF. In fact, the characteristics of the audio-visual objects (such as position, shape and color statistics) do not change much from frame to frame unless multiple sources with different activity patterns are present. Here we propose to keep the same segmentation procedure than for the first GoF while adding some priors (seeds) which are based on temporal consistency. In other words, we add a continuity prior on the audio-visual segmentation of the GoF.

We discussed before that the purpose of our particular division of the video signal into GoFs, in which neighboring GoFs share one frame, is to facilitate the smooth propagation of the knowledge forward in time. As explained in Algorithm 2, the regions in the shared frame which are labelled as foreground or background with enough confidence (their value in the trimap is either 1 or 0) are used to choose the *continuity seeds* that ensure temporal consistency between GoFs. In fact, we do not use all pixels with a clear label but only a subset of them, since our objective is to influence the segmentation without imposing a result. In our approach, the current audio-visual object is not

determined by a continuity prior but by the joint processing of audio and video signals, i.e. we give more importance to the *audio-visual seeds*.

The continuity seeds are chosen *randomly* from the set of pixels in the shared frame that are labelled as foreground and background in the trimap ($\mathcal{C}_f$ and $\mathcal{C}_b$). The number of continuity seeds is determined by

$$N_m^c = |\mathcal{C}_m| H_c \qquad \text{for} \quad m = \{f, b\} \tag{5.12}$$

where $|\mathcal{C}_m|$ denotes the cardinality of $\mathcal{C}_m$ and the parameter H_c controls the density of the continuity seeds in the shared frame. In all experiments we fix $H_c = 0.05$. Thus, the continuity seeds are composed by the 5% of pixels of the shared frame whose labels have enough confidence according to the trimap. The higher is H_c, the more continuity seeds we fix, and the more we rely on the prior information. If we decrease H_c we reduce the influence of segmentation result obtained for the previous GoF. Finally, $H_c = 0$ is equivalent to processing each GoF independently. Figure 5.4 [top right] shows the foreground and background continuity seeds in a real situation. As explained in Algorithm 2, the set of segmentation seeds in the current GoF is now composed of the continuity seeds in the first frame (shared frame) *and* the audio-visual seeds in the remaining frames. The audio-visual seeds are chosen as in the first GoF, that is following the procedure in Section 5.4.

Let us now discuss the color models estimation. As explained before, in the first GoF the color GMMs are learned on the audio-visual seeds. In the following GoFs, more information is available, since we know the color distributions of the audio-visual object and the background in the previous GoF. The first possibility is to learn the color models according to the last segmented GoF (or according to all previously segmented GoFs). This approach is counterproductive if multiple sources with different activity patterns are present (as in Figure 5.4). In this case, when only one source is active the foreground GMMs capture the color statistics of this source, while the color distribution of the other source (now inactive) is contained in the background model. Then, when the second source becomes active it is penalized by the color models, since the regional term in the segmentation in Section 5.3 links this audio-visual object to the background. Thus, the second source might not be successfully extracted even if the audio-visual seeds are correctly located over its video part. Furthermore, in this case the regional term links the first source to the foreground even if this source is not active any longer. The second possibility consists on using only the audio-visual seeds for the color models estimation. In this case, the color GMMs can change considerably from GoF to GoF, since the models consistency is not ensured. This could result on an unstable extraction of the audio-visual object, where the borders of the segmented regions are constantly changing within GoFs. As a result, a good compromise is to use both the continuity seeds and the audio-visual seeds in the estimation of foreground and background color models. In this way, when a new source becomes active, its colors are introduced in the GMMs computation by means of the audio-visual seeds. In addition, if the same source is active in two consecutive GoFs the borders of the segmented region will not change much since the color distribution of the source is also contained in the foreground GMM. Now, the question that arises is which proportion of continuity and audio-visual seeds is appropriate. Since we do not want the previous GoF's results to influence too much the segmentation of the current GoF, in all experiments the color GMMs are learned on a set of seeds composed of a 90% of audio-visual seeds and only a 10% of continuity seeds. In the experiments section we show the effect of varying the proportion of continuity and audio-visual seeds on the extracted regions when multiple sources alternate their periods of activity.

Finally, the segmentation of each GoF is obtained by applying the procedure described in Section 5.3 when using the *updated* seeds and color models, which contain information about the segmentation result in the previous GoF *and* the joint audio-visual analysis on the current GoF. At each step the extracted region is dilated and eroded to obtain a new trimap and reduce the risk of propagating

segmentation errors forward in time.

Notice that in some situations the joint audio-visual processing contradicts the temporal continuity prior, i.e. the audio-visual analysis shows that the source activity changes and the extracted region needs also to be modified consequently. Figure 5.4 shows the segmentation results in a period in which two audio-visual sources are present: the first one (right person) passes from active to inactive, while the other source (left person) does the opposite. In this case, the left person (active source in this GoF) is extracted successfully even though the continuity seeds in the first frame [top right] link it to the background. In contrast, the right person (now inactive) is still extracted in this GoF due to these continuity seeds. As explained before, in our set up the continuity priors affect the result but do not determine it (otherwise the new speaker could not be extracted). In fact, there needs to be a balance between the amount of information that we use from the temporal consistency and from the audio-visual analysis. The more knowledge we transfer from the result on the previous GoF, the longer a source that passes from active to inactive is extracted. However, the less we rely on the continuity priors, the more unstable are the results that we obtain, leading to extracted regions whose boundaries present high variations from GoF to GoF. This compromise is further discussed in the experiments section.

To summarize, in this section we have presented a segmentation procedure that divides the sequences into groups of frames (GoF) in order to process them sequentially but not independently. The knowledge that is extracted on a GoF is introduced in the next one in order to provide temporal continuity to the segmentation results and use the prior information that is available. However, in our approach the joint audio-visual processing has more weight in the extraction of the audio-visual object than the knowledge about the active source in the previous GoF. Thus, the extracted region is affected but not determined by the previous segmentation result. The proposed propagative procedure is computationally inexpensive, intuitive and effective.

5.6 Experiments

This section is divided into two parts. In Section 5.6.1 we present the extracted audio-visual objects in fragments of sequences, that is the segmentation results when analyzing *one* GoF. Thus, this set of experiments validates the first part of this chapter: Sections 5.2, 5.3 and 5.4. Then, Section 5.6.2 shows the results obtained on entire video sequences and validates the entire scheme for the extraction of audio-visual sources detailed in Section 5.5. Our audio-visual segmentation approach is demonstrated in challenging sequences presenting different types of sources, distracting moving objects and multiple sources with different activity patterns. A comparison between our method and previous audio-visual segmentation approaches in [40, 41] is also provided in this section.

For this purpose we use clips belonging to the *groups* section of the CUAVE database [62], and movies from two state-of-the-art source localization approaches presented by Monaci and Vandergheynst in [54] and Kidron et al. in [36]. Furthermore, another sequence recorded in a realistic environment is introduced to test complementary aspects of our approach.

The average processing time of automatically segmenting a video frame in a MacBook Pro laptop machine with an Intel Core 2 Duo CPU at 2.4 GHz and 2GB memory is about $2.5s$: $1.6s$ for the selection of audio-visual priors and $0.9s$ for the graph cut segmentation procedure. However, the diffusion process that is needed to determine the priors has not been optimized for the moment. It is currently coded in MATLAB and thus the processing time required for the choice of the segmentation seeds can drop drastically when parallelized. Notice that in the diffusion discretization described in [46] the value of the video signal at each point only depends on its six spatio-temporal neighbors.

Let us now briefly summarize the main parameters in the proposed audio-visual segmentation approach.

- Audio-visual diffusion parameters. We fix them in all experiments as explained in Chapter 4.
- The weights corresponding to the regional and audio-visual terms in the energy to minimize in Section 5.3: λ_R and λ_C respectively. These values define the relative importance of the color statistics and the audio-visual coherence with respect to the boundaries in the video signal. In all experiments we fix $\lambda_R = 0.05$ according to the range defined by [39] and [68], and $\lambda_C = 0.6$ in order to respect strong edges in the image. However, the results presented in this section do not change significantly for $\lambda_C \in [0.5, 1.5]$ and $\lambda_R \in [0.04, 0.06]$. The effect of varying this parameters on a real sequence is shown further in this section.
- The weight L of the links between the seeds and the corresponding (background or foreground) terminal. In all experiments we fix $L = \max_{p \in \mathcal{P}^j}(V_{p,q} + \lambda_C C_{p,q})$ as explained in Section 5.3. This value allows our segmentation procedure to change the seeds labels if required (notice that the seeds choice is unsupervised in our case). The higher is L the most difficult is that these labels are changed.
- The parameters H_f and H_b that determine the number of audio-visual seeds in each GoF. We fix $H_f = H_b = 3 \cdot 10^{-3}$ in all experiments, i.e. the 0.3% of pixels in the GoF are automatically labelled for foreground and background according to the procedure described in Section 5.4. Higher values for H_f and H_b could result in the choice of pixels that do not belong to the audio-visual object when distracting motion is present. Our algorithm selects the pixels with the highest audio-visual coherence as foreground seeds. Thus, our method chooses first pixels in the audio-related region, but after some point pixels belonging to the distractor can also be selected. Some examples of this situation in real sequences are provided further in this section.
- Number N_t of frames that compose a GoF. In all experiments $N_t = 20 - 25$ frames depending on the sampling rate of the analyzed sequence (the GoFs are around 1 second long). This choice is motivated for two main reasons. First, the hardware restrictions make it impossible to segment long time intervals because the number of vertices in the graph would be huge. Second, it is difficult to extract an audio-visual object for only a part of the GoF, since the regional and boundary terms link together homogeneous regions presenting similar color statistics. When one source passes from active to inactive within a GoF, its video part will be segmented for the entire GoF since many foreground seeds and very few background seeds will be situated over this audio-visual object (no background seeds are fixed in silent periods). Here we prefer to process shorter GoFs (N_t small) so that the extracted region changes faster in the transitions from active to inactive or vice versa.
- The parameters controlling the influence that the previous GoF segmentation result has in the current GoF: the ratio of pixels with clear labels in the shared frame that are used as continuity seeds H_c, and the proportion between continuity seeds and audio-visual seeds in the color models estimation. They are fixed in all experiments as suggested in Section 5.5: $H_c = 0.05$ and only a 10% of continuity seeds are used in the GMMs estimation. Those values ensure that the extracted region is influenced *but not determined* by the previous segmentation result. Higher values would lead to an audio-visual segmentation approach prevailing coherence between neighboring GoFs, while decreasing them would result on a reduction of the influence of the continuity prior. The limit, when $H_c = 0$ and no continuity seeds are used in the GMMs estimation, is equivalent to analyzing each GoF separately. The effect of using different proportions of audio-visual and continuity seeds for the GMMs is shown further in this section.

(a) Audio-visual coherence

(b) Foreground seeds - Result $H_b = H_f = 10 \cdot 10^{-3}$

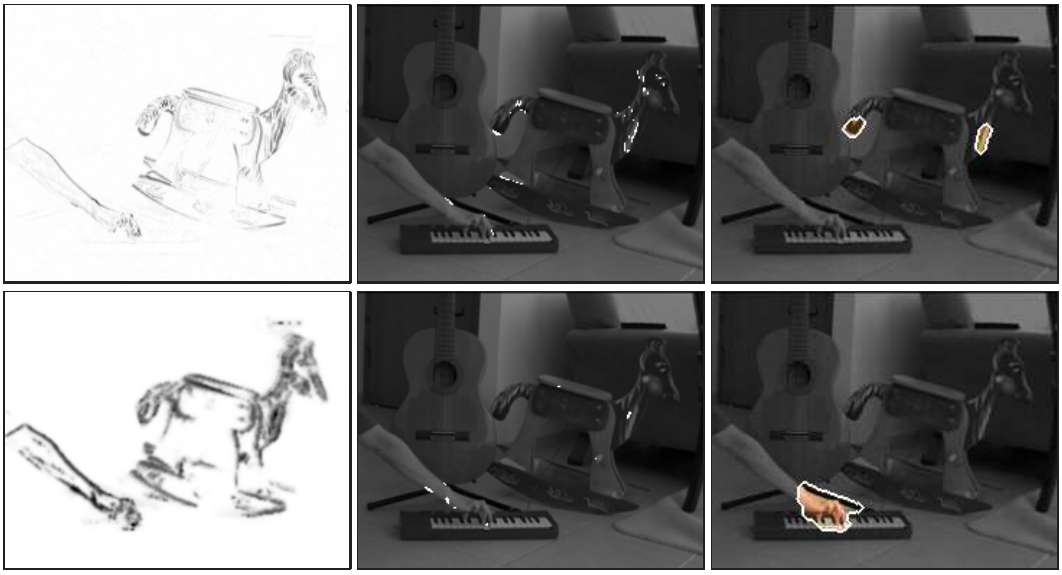
(c) Foreground seeds - Result $H_b = H_f = 3 \cdot 10^{-3}$

Figure 5.5 – *Audio-visual objects extracted by our method in two sequences containing distracting video motion when varying number of initial seeds. The background seeds are not depicted in this figure, but they are randomly distributed in the GoF.*

5.6.1 Results on *One* Video GoF

The proposed audio-visual segmentation algorithm is evaluated in fragments of sequences containing non-stationary sources, distracting moving objects and multiple simultaneous sources. Each video fragment is around 1 second length.

First, we show the segmentation results when analyzing two sequences containing a strong distracting motion (see Figure 5.5). The first clip (MovieA) [top] belongs to the source localization approach in [36], and it features a hand playing a synthesizer (non-stationary sound source) and a wooden rocking horse moving in the background. The second sequence [bottom] is a synthetic sequence composed of fragments of clips g01 and g08 from the groups partition of the CUAVE database in which two persons are present: the left person is uttering some numbers and the right one is mouthing the same numbers. Thus, both sequences are composed of a moving object associated to the audio signal (hand and left person) and another one that represents a strong visual distraction, whose motion is either periodic [top] or very similar to the motion in the audio-visual object [bottom]. In Figure 5.5 we show the effect of varying the initial number of seeds on the extracted region when there is distracting motion in the scene. If we use a very small number of



(a) Features (b) Foreground seeds (c) Results

Figure 5.6 – *Extracted audio-visual objects when the segmentation seeds are chosen according to the original motion in the sequence [top] and the estimated audio-visual coherence [bottom].*

seeds (c) as suggested in Section 5.4, the audio-visual object is successfully determined for both clips and the extracted region does not contain the distractor. In this case, few seeds are wrong (located over the horse or the right person) because only the 0.3% of pixels in the GoF are initially labelled. However, when H_f and H_b increase drastically (in (b) we have 1% of seeds), the number of foreground seeds located over the distractor grows too and the extracted region can contain parts that do not belong to the audio-visual object. As a result, in (b) a big part of the rocking horse is extracted [top] and the mouth of the wrong person is segmented in some frames too [bottom].

MovieA is also used to illustrate the necessity of choosing the highest values of the audio-visual coherence in Section 5.2 as the priors for the audio-visual segmentation process. Figure 5.6 compares the extracted region when using the audio-visual coherence [bottom] to the result obtained when using simply the motion in the original video sequence [top]. In this case the motion is generated by the distractor is more intense than the motion in the audio-visual object. Thus, when choosing the foreground seeds according to the original motion [top], the extracted region contains only parts of the rocking horse since most of the seeds are located over this distracting moving object. In contrast, the highest values of the audio-visual coherence are correctly situated over the hand playing the synthesizer and, as expected, the audio-visual object is successfully extracted in the bottom row.

Next, we provide a qualitative comparison between our method and previous audio-visual segmentation approaches in [40, 41]. For this purpose, we use several fragments of sequences g22 and g23 of the CUAVE database, where two and three persons speak in turns respectively. The objective of this experiment is to illustrate the advantages of the proposed method in real situations. Figure 5.7 shows the segmentation results obtained by the approaches in [40, 41] in the top row and the audio-visual objects extracted by our method in the bottom row. The foreground seeds in this frames are depicted in the middle row. Our results are specially favorable in (c): our method extracts the entire mouth region, while the approach in [40] segments mostly the speaker's hair. Furthermore, in (e) the entire girl's face is extracted by our approach [bottom], while only the

(a) (b) (c) (d) (e)

Figure 5.7 – *[Top] Extracted regions when applying the method in [40] to sequence g23 [left] and the approach in [41] to movie g22 [right]. [Middle] Foreground seeds chosen according to the audio-visual coherence. [Bottom] Results when using our audio-visual segmentation approach. In all situations the current speaker is detected.*

mouth region can be extracted in [41] [top]. As discussed before, the audio-visual term proposed by the previous methods in [40, 41] penalizes the extraction of pixels presenting a low coherence with the soundtrack. In contrast, our audio-visual term does not affect these regions and the presence of the regional term in our case facilitates the extraction of regions homogeneous in color. Thus, our method seems more suitable for applications that require the entire face region of the current speaker (or a more complete source region in general). An example of such an application can be the protection of the speaker's identity by automatically mosaicing his/her face.

Sequence g23 of the CUAVE database is also used to illustrate the effect of varying the weights of regional and audio-visual terms in equation 5.2 (λ_R and λ_C respectively) on the extracted audio-visual objects. The results obtained for three different GoFs in this sequence are depicted in Figure 5.8 (in each GoF a different person is speaking). In (b) both weights are zero, in (c) and (d) we only consider the regional term and the audio-visual term respectively, and finally in (e) we show the results when both terms are taken into account. The unsupervised choice of the segmentation priors is highly accurate, leading to a concentration of foreground seeds in the mouth region of the current speaker in (a). As a result, the speaker's mouth is already extracted when using only the boundary term in equation 5.2. When adding the regional term in (c) the extracted region becomes broader, since the mouth label is spread across the face region due to its homogeneous color. The audio-visual term links together the pixels in the mouth of the current speaker, since in this region the audio-visual coherence is high. This effect can be observed in the three analyzed cases. In the middle frame, when the regional term is not considered, the audio-visual term allows the chin extraction in (d) while only the lips were segmented in (b). In the top and bottom frames, when comparing (d) and (e) we observe that a broader face region is extracted when adding the audio-visual term in the energy function. Finally, we can conclude that using both audio-visual and regional terms (e) leads to the extraction of a more complete audio-visual object in all cases.

A final experiment is performed on a fragment of clip g21 of the CUAVE database in which the two persons in the camera field of view speak at the same time. Our purpose in this case is to demonstrate that our audio-visual segmentation method is able to extract multiple simultaneous sources. The results obtained in this sequence are depicted in Figure 5.9. In some of the frames

(a) Foreground seeds | (b) Results $\lambda_R = 0, \lambda_C = 0$ | (c) Results $\lambda_R = 0.05, \lambda_C = 0$ | (d) Results $\lambda_R = 0, \lambda_C = 0.6$ | (e) Results $\lambda_R = 0.05, \lambda_C = 0.6$

Figure 5.8 – *Results when varying the parameters λ_R and λ_C for three different GoFs of sequence g23. [From top to bottom] The left, center and right persons respectively are speaking in the depicted frames.*

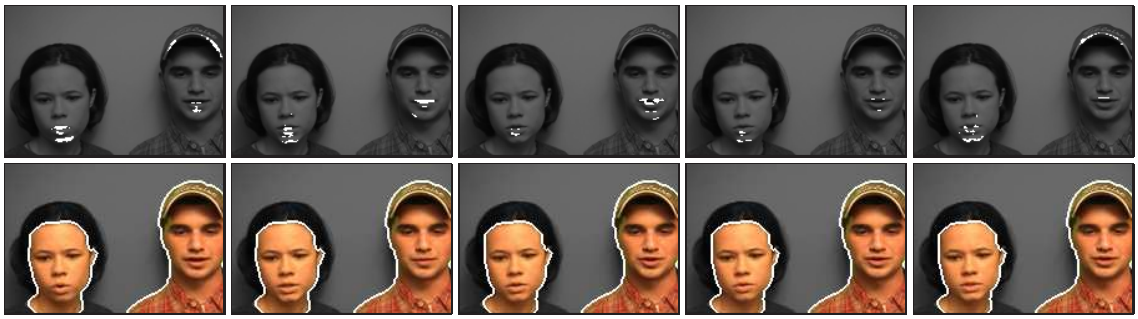


Figure 5.9 – *Results on a fragment of sequence g21 of CUAVE database in which two persons speak simultaneously. The foreground seeds in these frames are depicted on the top row, while the results appear in the bottom row.*

the foreground seeds are mainly situated over the left person, while in other frames most seeds are located over the right person. However, in average the seeds are located over the mouth regions of both speakers most of the time. As expected, the extracted region contains the faces of the two speakers, since both of them represent audio-visual objects in the scene.

Several experiments have been performed to evaluate the performance of the proposed audio-visual segmentation approach when analyzing *one* GoF. The procedure described in Sections 5.2, 5.3 and 5.4 has been verified under different conditions including several types of sources, distracting moving objects and multiple simultaneous sources. We have shown the effect of the parameters on the extracted audio-visual objects, and the efficiency of the audio-visual coherence as a measure of the correlation between video structures and the sounds. Finally, we have also demonstrated the advantages of our method over the approaches in [40, 41]. Our audio-visual segmentation approach is able to extract more complete audio-visual objects because our method does not force regions with a lower audio-visual coherence to be labelled as background.

5.6.2 Results on Entire Video Sequences

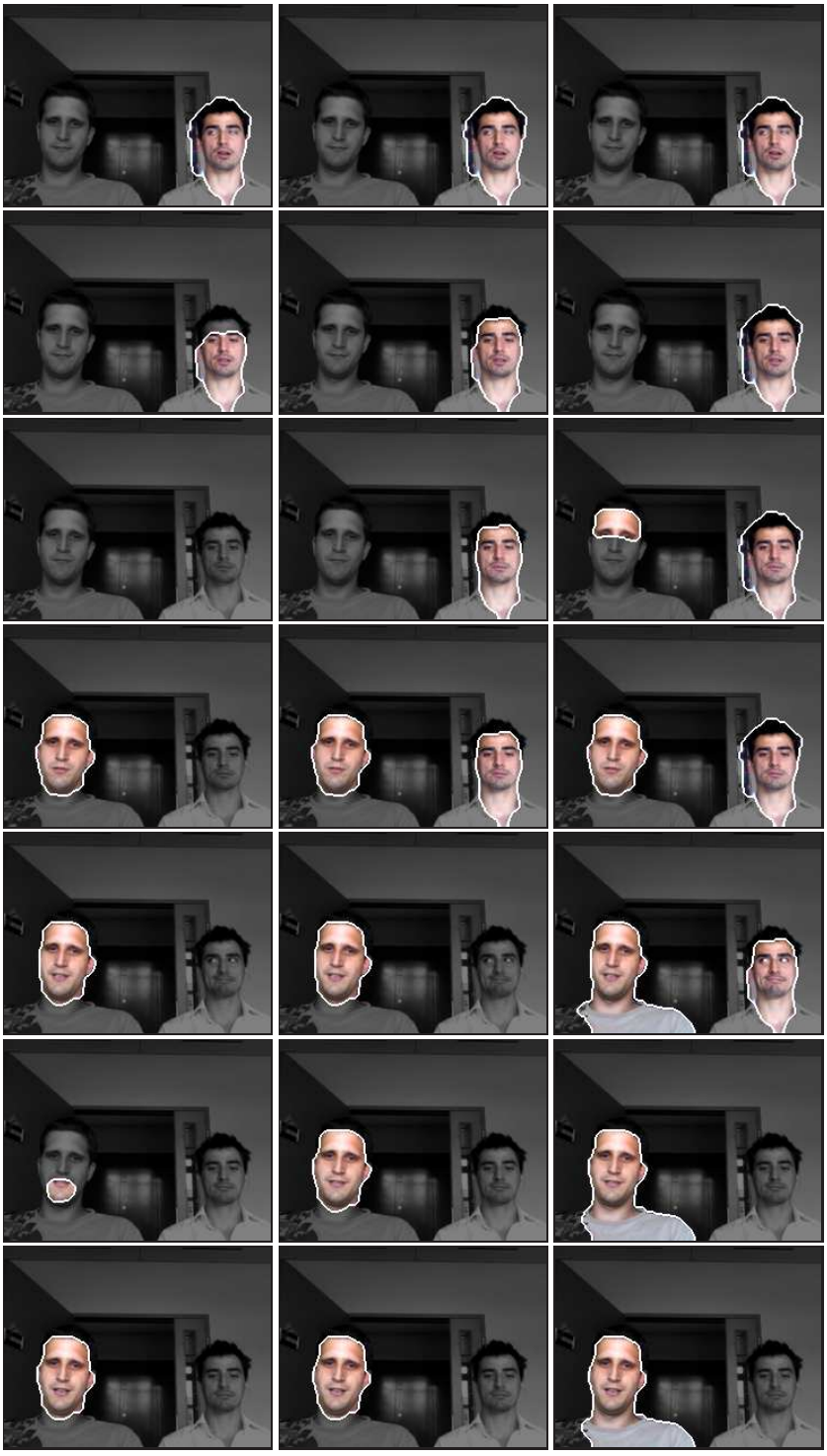
The following set of experiments evaluate the algorithm that was proposed in Section 5.5 for the propagation of the segmentation results forward in time. For this purpose, we analyze longer video sequences that contain non-stationary sources, distracting video motion and multiple audio-visual sources with different activity patterns. Here we test our method when the sources pass from active to inactive or vice versa. Our propagative segmentation approach is really challenged in this case since the transfer of the information from one GoF to the next one can be counterproductive.

Videos showing the original and segmented sequences obtained with the proposed method are available online at `http://lts2www.epfl.ch/~llagoste/AVobjectExtraction_results.htm`.

The first experiment tries to provide a deeper understanding on the effect of relying more or less in the previous GoF segmentation result when estimating the color models of the sources in the current GoF. For this purpose, we run three times the simulations on the same sequence with different percentages of audio-visual seeds and continuity seeds. The analyzed sequence is composed by two persons speaking in turns. Unlike in clips from CUAVE database, the speakers are not situated in front of a green flat background but in a realistic office environment. This movie is recorded with an iSight camera integrated into a MacBook Pro laptop at 25 frames/sec with a resolution of 640×480 pixels which is resized to 240×180 pixels for its analysis. The length of the sequence is around $9s$.

The results for several frames of this sequence can be observed in Figure 5.10. In (a) the color models are learned on the audio-visual seeds only, in (b) we introduce a 10% of continuity seeds and in (c) the continuity seeds represent the 50% of pixels used in the GMM computation. Notice that *in all cases the segmentation seeds include the audio-visual and continuity seeds*, i.e. the percentages considered in this analysis only affect the GMM estimation. Results show that the borders of the extracted object are very unstable when using only the audio-visual information in (a). In fact, when the audio-visual seeds are highly concentrated in the mouth region, the foreground GMM captures the color distribution in this region and, as a result, the rest of the face may not be extracted. Some examples can be observed in (a) rows two and six, where the extracted region is smaller than in the previous GoF and it does not contain the entire speaker face. Thus, to improve the segmentation stability, some continuity seeds need to be introduced in the GMMs estimation to profit from the knowledge about the color statistics in the previous GoF. However, we should be careful because if the color models rely too much on the prior information we will force the extraction of the same region even if the source is not currently active. This is evident when looking at Figure 5.10(c), the same number of audio-visual seeds and continuity seeds are used in the GMM estimation. In this case the right person is extracted for a considerably long period after becoming inactive (there are 2 seconds approximately between rows three and five). A satisfactory compromise is reached in (b) when fixing the proportion between audio-visual and continuity seeds as suggested in Section 5.5. In this case, the few continuity seeds that are used in the GMMs estimation ensure the stability of the audio-visual object borders in (a), and the extracted region does not contain the right person less than one second after he stops speaking.

Several experiments are conducted on sequences presenting distracting motion in the camera's field of view. Figures 5.11 and 5.12 show several frames belonging to three sequences in which a hand is playing a piano. The three sequences are taken from state-of-the-art audio-visual source localization approaches: the first two movies belong to the work presented by Monaci and Vandergheynst in [54], while the third movie was used by Kidron et al. in [36]. The distracting motion is generated by a toy car crossing the image in Figure 5.11(a), a fan in Figure 5.11(b) and a rocking wooden horse in Figure 5.12. Thus, the distracting motion is sporadic in the first case, continuous



(a) 100% AV seeds
0% continuity seeds

(b) 90% AV seeds
10% continuity seeds

(c) 50% AV seeds
50% continuity seeds

Figure 5.10 – *Effect of introducing the knowledge from last GoF's segmentation result in the color models estimation when two sources alternate their periods of activity. In (a) only seeds from audio-visual processing are used in the GMM computation, in (b) a 10% of continuity seeds are introduced, and in (c) the continuity seeds represent a 50%. [From top to bottom] At the beginning only the left person is speaking, then he stops and the right person starts speaking. The transition is produced between third and fourth rows.*

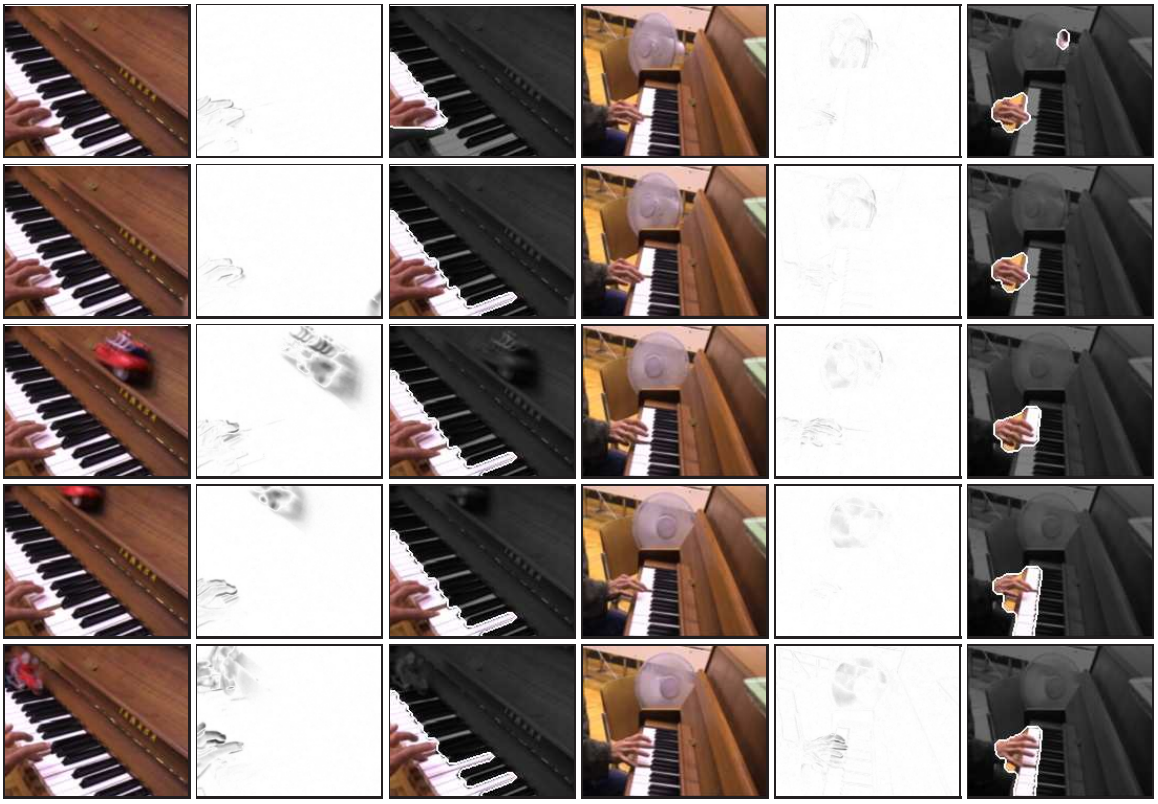


Figure 5.11 – *Extracted audio-visual objects in the presence of distracting motion, which is generated by a toy car in (a) and by a fan in (b).*

in the second movie and periodic in the third sequence. In the first two sequences (in Figure 5.11) the extracted region contains the audio-visual object (hand) all the time. In fact, the segmented region is composed mainly of the hand region in the first frames, while it also contains the entire keyboard towards the end. This behavior is normal since when the keys are pressed, their motion is synchronized with the audio signal. Furthermore, they represent a very homogeneous region in terms of color. Notice that the black keys are not extracted because they are not pressed at any time. Let us now discuss the results obtained on the third sequence (in Figure 5.12). This represents the most challenging situation since the distracting motion is much more intense than the motion generated by the audio-visual object. In this case the audio-visual object can not be extracted in the last part of the sequence. The main reason is that in the final frames the motion in the hand region is very small compared to the horse motion. As a result, the foreground audio-visual seeds are divided between both regions and there is not enough concentration of seeds around the hand to allow its extraction. Notice however that the distracting moving object is not contained at any time in the extracted region. In the last period our method fails to extract the audio-visual object but it does not make an error by extracting the wrong object (distractor).

Finally, Figure 5.13 illustrates the main limitations of our approach. The analyzed sequence corresponds to a fragment of clip g14 of CUAVE database in which two persons speak in turns, first the left one and then the right one. The current speaker in the depicted frames is always correctly detected. However the entire face region is not extracted for the left speaker (see the first two

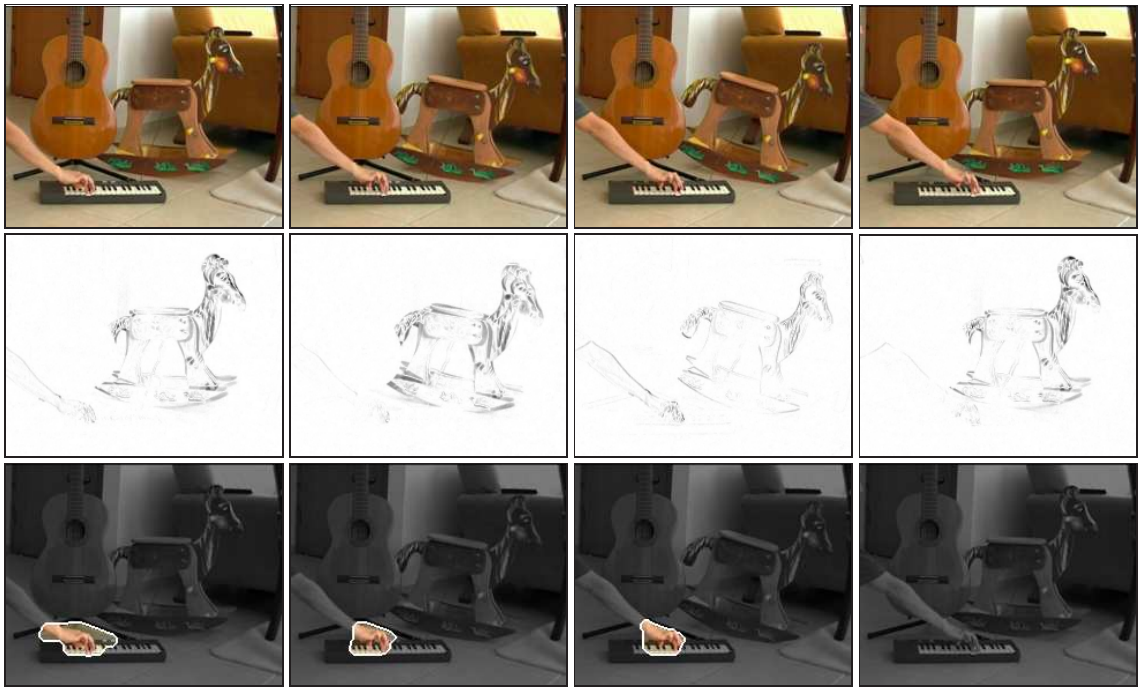


Figure 5.12 – *Extracted audio-visual object when the sound source is non-stationary and a strong distracting motion is present. [From top to bottom] Video frame, motion and extracted audio-visual object.*

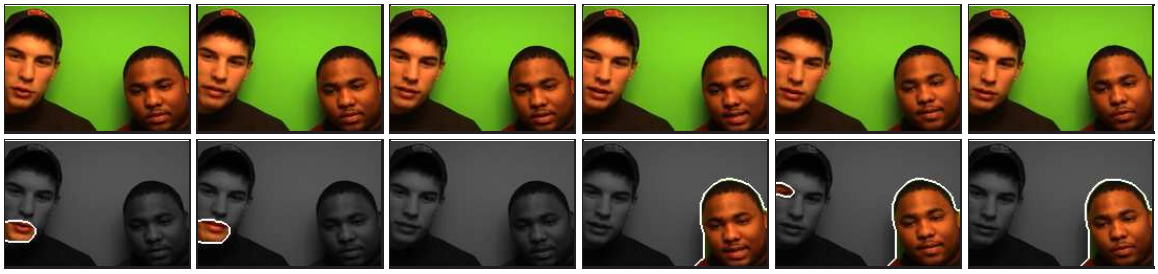


Figure 5.13 – *Results obtained for a fragment of sequence g14 of the CUAVE database where two persons speak in turns. The third column corresponds to the silence period between the speech corresponding to the left person and sounds generated by the right person.*

frames). In fact, since the proposed audio-visual segmentation approach is unsupervised we do not have control over the segmented region. Remember that the purpose of our method is to extract automatically the audio-visual objects that are present in a scene. Results show that this objective is achieved by our approach with a good accuracy. As a final remark, notice that small and sporadic artifacts are extracted in some cases, as the left person's eye in the fourth frame of Figure 5.13 or a fragment of the fan in the first frame of Figure 5.11(b). This artifacts can easily be removed by eroding/dilating the segmented region both in space and time. Following this simple procedure all small regions extracted during a short period of time can be efficiently eliminated.

Table 5.2 provides a quantitative analysis of the results in this section. Two quantities are used for this purpose: the *detection rate* and the *misdetection rate*. The *detection rate* measures the percentage of time in which the active audio-visual source is extracted. Thus, the higher is this value the more efficient is our algorithm in segmenting the audio-visual objects in the scene.

Sequence	Detection Rate (%)	Misdetection Rate (%)
Fig. 5.10(b)	96	13
Fig. 5.11(a)	100	0
Fig. 5.11(b)	100	8
Fig. 5.12	59	0
Fig. 5.13	87	10
MEAN	90	8

Table 5.2 – *Detection and misdetection rates for the sequences analyzed in this section. The mean values are computed by taking into account the number of frames composing each sequence.*

In contrast, the *misdetection rate* measures the percentage of time in which the extracted region contains an incorrect part of the image, that is either a visual distractor or an inactive audio-visual source. As a result, this second quantity measures the presence of errors in the segmented region. Notice that the addition of both quantities is not necessarily 100%, since our algorithm can extract the active audio-visual source *and* some incorrect image region in the same frame. The *detection rate* and the *misdetection rate* are independent measures, and by combining them we can quantify our method's ability in extracting the active source without extracting at the same time distracting moving objects or inactive sources. Given the different lengths of the analyzed movies, the number of frames that compose each sequence are considered for the computation of the mean values for the detection and misdetection rates in Table 5.2.

Let us now discuss in detail this quantitative evaluation. In this section we have applied our method to five sequences depicting challenging situations such as multiple audio-visual sources alternating their periods of activity or distracting moving objects in the camera's field of view. In the 90% of analyzed frames the current active audio-visual source is successfully extracted. The remaining frames (in which the source is not detected) represent thus the 10% of frames and they can be divided into two main groups. The first group represents the 47% of undetected cases, and it is composed of frames which are situated in the transitions between the sources' activity periods (when they pass from inactive to active or vice versa). An example of this behavior can be found in the sequence depicted in Figure 5.13, where the left speaker's mouth (active source) is not detected for a short period before he stops speaking. The second group (53% of undetected cases) is composed by the final frames of the sequence in Figure 5.12, in which the audio-visual object (hand) can not be extracted due to the magnitude of the distracting motion generated by the rocking horse. As discussed before, in this case the concentration of foreground seeds in the hand region is not high enough to allow its extraction.

Concerning the *misdetection rate*, a part of the image which does not belong to an active audio-visual source is extracted by our method in 8% of frames. A fragment of a video distractor is extracted in 54% of misdetections, such as the small part of the fan at the beginning of the sequence in Figure 5.11(b) and the inactive speaker's eye in Figure 5.13. The rest (46% of misdetections) corresponds again to transitions between the sources' activity periods, e.g. the extraction of the left person (inactive source) in Fig. 5.10(b) for a short period after he stopped speaking. As explained before, the first group of misdetections can be removed with a simple post-processing step penalizing small regions that are extracted for a short period. In contrast, the misdetections in transitions between the sources' activity periods are difficult to eliminate since they result from

the division of the signals into GoFs and the propagation of the segmentation results forward in time. However, in all cases the delay between the moment in which a source becomes active and its extraction is small (less than one second), and the same happens when the sources become inactive.

To summarize, the proposed method provides a good accuracy in the extraction of the active audio-visual sources in the scene, by combining a high *detection rate* with a low *misdetection rate*. Furthermore, we can expect our method to improve its performances in both quantities when analyzing sequences representing less challenging situations.

5.7 Discussion

In this chapter, we have presented a novel method to automatically extract the audio-visual objects present in a scene. First, the correlation between the sounds and the motion in the video signal is assessed. Video regions presenting high audio-visual coherence are used as the starting point for a graph cut segmentation procedure whose goal is to extract the video modality of the source. The proposed approach uses the knowledge obtained from joint audio-visual processing in the unsupervised selection of the segmentation priors and in the energy term that the graph cuts minimize. Furthermore, an intuitive and computationally inexpensive propagation algorithm is introduced to allow the extraction of audio-visual objects in longer sequences while preserving the spatio-temporal continuity of the result.

Our approach has been tested in challenging sequences containing distracting moving objects and different types of audio-visual sources. In all cases the video modality of the source has been successfully extracted. Our definition of the segmentation problem, which includes an audio-visual term *and* a regional term encouraging homogeneous regions, makes our method suitable for applications that require the extraction of complete audio-visual objects. For example the whole speaker's face region might be needed when trying to protect the speaker's identity by automatically mosaicing his/her face. Regarding the extraction of audio-visual objects in longer video sequences, the proposed propagative procedure has been successfully applied to sequences presenting distracting motion and multiple sources with different activity patterns. The extracted regions are stable and they evolve according to the changes in the sources activity in a short time delay. Our approach is able to distinguish between real audio-visual objects and distracting moving objects, leading to a good accuracy in the extraction of regions related to the sounds. Since the segmentation method that we have presented in this chapter is unsupervised we do not have control over the extracted region. As a result, our approach is able to extract the region whose motion is synchronous with the sounds but this region might not fit the user expectations. For example, the user might want to extract the entire guitarist body instead of only his hand or arm to compose it into a new background. For this kind of applications a semi-supervised approach allowing the user to add some more seeds and determine the extracted region could be more appropriate. This is the object of future research and it is further discussed in the conclusions chapter.

6 Conclusion

6.1 Discussed Topics and Achievements

As explained in the introduction, many fusion methods have been proposed in the last decade to combine the information captured by one video-camera and one microphone. All these methods are based on an assumption of synchrony between related events in audio and video channels. Their purpose is to identify the relationships between moving objects and the sounds that they generate. Then, this information is used in numerous applications, such as the spatial localization and tracking of sound sources, speech recognition, speech enhancement, sound source separation, emotion recognition, automatic music transcription and video classification. In most cases the fusion between audio and video modalities is performed by following the same strategy: first define simple features for each modality separately and then combine them in a fusion step which is based on canonical correlation analysis or the estimation of joint probabilities for audio and video features. In fact, many of these approaches assess the synchrony between each pixel's temporal variations and the soundtrack. Thus they assume implicitly that the pixels are independent conditioned on the audio signal. This is not true in general, and the results that they obtain are sensitive to noise and they do not ensure spatial consistency.

In this thesis we have presented two novel audio-visual fusion methods which are based on completely different strategies. The first approach is focused on the modeling of audio and video signals while the second one is concentrated on the fusion step to combine them. However, both methods exploit the spatio-temporal consistency that characterizes video signals: we *do* assess the synchrony between *moving regions* and sounds.

Most approaches in this domain use low-level features for audio and video signals and, as a result, it is difficult to connect these features with the physics of the problem. The goal behind audio-visual fusion is to assess the synchrony between moving image structures and sounds, and thus the changes that an isolated pixel experiments do not have a real meaning. According to this observation, the fusion method in Chapter 2 decomposes first audio and video signals into a set of basic structures having a true physical meaning, which are respectively a concentration of energy in

the time-frequency plane (sound) and a geometric image structure and its temporal evolution (a 2D projection of an object whose position with respect to the camera is changing for example). Then, meaningful events for audio and video modalities are defined as the presence of a sound and the motion of a salient image structure. The fusion step in this case is very simple since we only need to evaluate the co-ocurrence between these events.

This fusion approach has been applied in Chapter 3 to the localization and separation of the audio-visual sources that compose a scene. Video structures presenting a strong correlation with the audio signal are grouped together using a clustering algorithm that is able to count and locate the sources in the image. The video part of each source is reconstructed by adding the contribution of all the atoms composing the source. Next, temporal periods in which the sources are active alone are detected and used to learn models characterizing their acoustic frequency behavior. Finally, the contribution of each source to the soundtrack is also separated in mixed periods.

At this point of the thesis the reader is already aware that there is plenty of information in the video signal which is not needed for the joint processing of audio and video modalities. For example background regions or objects whose motion is not related to the sounds do not help us in tasks such as speech recognition or sound source localization. This motivation is behind the fusion method in Chapter 4, which uses a PDE-based nonlinear diffusion approach to remove from the video signal all information that is not required for applications in this domain. For this purpose, we have defined an audio-visual diffusion coefficient which is an estimate of the synchrony between video motion and audio energy (sounds). The proposed diffusion procedure erodes progressively the video signal and converts it into an audio-visual video signal containing only the information in regions whose motion is coherent with the soundtrack. Notice that we consider regions and not pixels, since the 3D characteristic of our audio-visual diffusion approach ensures spatio-temporal consistency by prevailing *image structures* moving coherently with sounds. The regions that are better preserved through the diffusion procedure are thus likely to be part of an audio-visual object, that is the video modality of an audio-visual source.

According to this observation, in Chapter 5 we have proposed an unsupervised segmentation approach based on graph cuts whose objective is to extract the audio-visual objects that are present in a scene. Our segmentation method is designed to keep together pixels in video regions presenting high audio-visual coherence. The initial information about foreground and background locations that is required to start the segmentation process is provided by the audio-visual diffusion procedure in the previous chapter. Finally, a propagative scheme that transfers the segmentation results forward in time is proposed to deal with longer video signals. Our approach has been successfully applied to sequences presenting moving sources, strong distracting motion and multiple sources alternating their activity periods.

Even though the two audio-visual fusion methods in this thesis are completely different, we have demonstrated that both approaches are able to combine efficiently audio and video modalities. The main reason behind this good behavior is that both approaches exploit the spatio-temporal consistency that characterizes video signals, i.e. neighboring pixels have often similar characteristics because they are probably part of the same 3D structure in the real world. In all cases we assess the synchrony between moving image *structures* and sounds, and this represents a significant advantage over previous methods. Another strength of our fusion approaches is that they are completely general and, as a result, they can be applied to all kind of audio-visual sources. Some of the previous methods in this domain were focused on the analysis of sequences composed of speakers. Thus, they pre-selected the speaker's mouth region in order to extract specific video features or to learn the joint distributions of audio and video features. Let us stress that our approaches do not require any training procedure. Finally, the two fusion methods presented in this thesis can deal

with multiple simultaneous audio-visual sources while many of the previous approaches did not even consider that case.

6.2 Future Research Directions

After this thesis, several possible directions for future research can be considered. Let us now introduce them.

- The decomposition of a video sequence into redundant dictionaries of atoms in Chapter 2 has been conceived for video coding. Even if this representation has been demonstrated to be suitable for joint audio-visual processing, the entire video decomposition might not be required. Indeed many atoms that are used in the representation of the background (static regions) or the distracting moving objects are not used in the BAVSS algorithm in Chapter 3, since these atoms do not move synchronously with the presence of sounds in the audio channel. However, the 3D-MP algorithm that decomposes the video signal requires a high computational cost to extract those unnecessary atoms (and specially to track the 2D structures from frame to frame). In fact, we could use the audio-visual diffusion approach in Chapter 4 to pre-select the regions of interest in which the sources can be located. The 3D-MP algorithm could be modified in order to look for atoms in those regions by adding some priors in the MP algorithm (preconditioning the search). Another option could be to decompose the diffused video signal, where the information is concentrated around audio-related video regions. As a result, the first atoms in the 3D-MP decomposition would try to approximate the image structures that the diffusion procedure highlights. In both cases the signal could be decomposed into a much smaller number of relevant image structures located in pre-selected regions of interest. Thus, a significant amount of computational cost could be saved.

- From an application point of view, it could also be possible to use the second audio-visual fusion method in Chapter 4 for the separation of sources in audio and video modalities. Since the graph-cut based segmentation allows the interaction of the user, in this case we could design a graphical interface for the semi-supervised extraction of the sources. The user could select a part of the video and extract automatically the audio signal which is associated to this video region. For example we could decide which part of a musician we want to be extracted by adding seeds through the graphical interface. Then, our algorithm would detect the periods of activity of this audio-visual source and extract consequently the sounds that it generates. Once a source is extracted it could be composed into a new video and audio-visually mixed with other singers and/or music instruments. Thus, this could be the base for an audio-visual processing software allowing to compose videos of several singers and musicians extracted from different sequences. We have already started some preliminary work in this direction, which can be found in [14].

- After the fusion approach presented in Chapter 4 one question arises. In fact, we have used the nonlinear diffusion to create an *audio-visual* video signal from a video signal by combining audio and video channels. The diffused volume contains thus only the video information which is audio-visual, i.e. the image structures that are likely to belong to an audio-visual source. Then, why do not use the video signal to generate an *audio-visual* audio signal? In this case only the sounds that are likely to belong to an audio-visual source would be kept. As a video feature we could use the amount of motion in the video signal for example.
  In fact, by alternating between *audio-visual* video diffusion and *audio-visual* audio diffusion

we could remove the contribution of all sources that are not audio-visual. Typical audio-only sources are sounds that are generated by a moving object out of the camera field of view (e.g. another person speaking) or without any associated motion (e.g. music from a Hi-Fi equipment), while video distractors are constituted by any moving object that does not generate sounds. In this case, a first iteration of the diffusion procedure in Chapter 4 would remove video information (and motion) from regions that are not synchronous to the sounds. Next, an iteration of nonlinear diffusion *in the audio signal* (possibly on the 2D domain represented by the spectrogram) would eliminate the sounds that do not have an associated motion. As a result, a simple comparison of the frequencies attenuation would show which frequencies are characteristic of the audio-only source, which ones are occupied by the (one or more) audio-visual sources and which frequencies do they share (and in which proportion). Then, the contribution of the audio-only source could be eliminated from the soundtrack. This analysis is similar to the approach that allows us to extract the audio-visual objects in Chapter 5: by comparing the effect of the diffusion in each region, we determine the video regions that are less affected and use this information to extract the audio-visual objects.
Some examples of applications can be the removal of the public sounds and applause in the recording of a concert, or the attenuation of the cars' background noises when a person is recorded speaking in the street.

Bibliography

[1] T. Anderson (1984). *An introduction to multivariate statistical analysis*. Wiley series in probability and mathematical statistics. Wiley, New York, 2nd edn.

[2] S. Arberet, R. Gribonval, F. Bimbot (2010). A robust method to count and locate audio sources in a multichannel underdetermined mixture. *IEEE Trans. on Signal Processing* **58**(1):121 –133.

[3] G. Aubert, P. Kornprobst (2006). *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, vol. 147 of *Applied Mathematical Sciences*. Springer.

[4] X. Bai, J. Wang, D. Simons, G. Sapiro (2009). Video snapcut: robust video object cutout using localized classifiers. In *Proc. of ACM SIGGRAPH*, pp. 1–11.

[5] R. Baken, R. Orlikoff (2000). *Clinical Measurement of Speech and Voice*. Singular Publishing Group Thomson Learning, San Diego, CA, 2nd edn.

[6] Z. Barzelay, Y. Y. Schechner (2007). Harmony in motion. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[7] M. J. Beal, N. Jojic, H. Attias (2003). A graphical model for audiovisual object tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **25**(7):828–836.

[8] L. Benaroya, F. Bimbot (2003). Wiener based source separation with HMM/GMM using a single sensor. In *Proc. of Int. Symp. on Independent Component Anal. and Blind Signal Separation (ICA2003)*, pp. 957–961.

[9] S. Bengio (2003). Multimodal authentication using asynchronous hmms. In *Proc. of Int. Conf. on Audio- and Video- based Biometric Person Authentication (AVBPA'03)*, pp. 770–777.

[10] P. Besson, V. Popovici, J.-M. Vesin, J.-P. Thiran, M. Kunt (2008). Extraction of audio features specific to speech production for multimodal speaker detection. *IEEE Trans. on Multimedia* **10**(1):63 –73.

[11] BGL (2002). *The boost graph library: user guide and reference manual*. Addison-Wesley Longman Publishing Co., Inc., Boston, USA.

[12] Y. Boykov, M.-P. Jolly (2001). Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *Proc. of IEEE Int. Conf. on Computer Vision (ICCV)*, pp. 105–112.

[13] T. Butz, J.-P. Thiran (2005). From error probability to information theoretic (multi-modal) signal processing. *Signal Processing* **85**(5):875–902.

[14] P. Calatayud Martinez, A. Llagostera Casanovas, P. Vandergheynst (2010). Semi-supervised Extraction of Audio-Visual Sources. Master thesis, EPFL, [Online] Available: http://infoscience.epfl.ch/record/150838/.

[15] F. Catte, P. Lions, J. Morel, T. Coll (1992). Image selective smoothing and edge detection by nonlinear diffusion. *SIAM Journal on Numerical Analysis* **29**(1):182–193.

[16] S. S. Chen, D. L. Donoho, M. A. Saunders (1998). Atomic decomposition by basis pursuit. *SIAM Journal of Scientific Computing* **20**(1):33–61.

[17] G. Chetty, M. Wagner (2006). Audio-visual multimodal fusion for biometric person authentication and liveness verification. In *Proc. of NICTA-HCSNet Multimodal User Interaction Workshop (MMUI '05)*, pp. 17–24.

[18] G. Chetty, M. Wagner (2007). Audio visual speaker verification based on hybrid fusion of cross modal features. In *Pattern Recognition and Machine Intelligence (PReMI)*, pp. 469–478.

[19] R. Cutler, L. Davis (2000). Look who's talking: speaker detection using video and audio correlation. In *Proc of. IEEE Int. Conf. on Multimedia and Expo (ICME)*, vol. 3, pp. 1589 –1592.

[20] R. Dansereau (2004). Co-channel audiovisual speech separation using spectral matching constraints. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 5, pp. 645–648.

[21] G. Davis, S. Mallat, M. Avellaneda (1997). Adaptive greedy approximations. *Journal of Constructive Approximations* **13**(1):57–98.

[22] S. Deligne, G. Potamianos, C. Neti (2002). Audio-visual speech enhancement with AVCDCN (Audiovisual Codebook Dependent Cepstral Normalization). In *Proc. of Int. Conf. Spoken Language Processing (ICSLP)*, pp. 1449–1452.

[23] O. Divorra Escoda, G. Monaci, R. Figueras i Ventura, P. Vandergheynst, M. Bierlaire (2009). Geometric video approximation using weighted matching pursuit. *IEEE Trans. on Image Processing* **18**(8):1703–1716.

[24] J. Driver (1996). Enhancement of selective listening by illusory mislocation of speech sounds due to lip-reading. *Nature* **381**(6577):66–68.

[25] R. Fendrich, P. M. Corballis (2001). The temporal cross-capture of audition and vision. *Perception & Psychophysics* **63**(4):719–25.

[26] J. W. Fisher, T. Darrell (2004). Speaker association with signal-level audiovisual fusion. *IEEE Trans. on Multimedia* **6**(3):406–413.

[27] J. Fritsch, M. Kleinehagenbrock, S. Lang, G. A. Fink, G. Sagerer (2004). Audiovisual person tracking with a mobile robot. In *Proc. of Int. Conf. on Intelligent Autonomous Systems*, pp. 898–906.

[28] O. Gillet, G. Richard (2005). Automatic transcription of drum sequences using audiovisual features. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 3, pp. 205–208.

[29] L. Girin, J.-L. Schwartz, G. Feng (2001). Audio-visual enhancement of speech in noise. *Journal of the Acoustical Society of America* **109**(6):3007–3020.

[30] R. Goecke, G. Potamianos, C. Neti (2002). Noisy audio feature enhancement using audio-visual speech data. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, pp. 2025–2028.

[31] I. F. Gorodnitsky, B. D. Rao (1997). Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm. *IEEE Trans. on Signal Processing* **45**(3):600–616.

[32] M. Gurban, J.-P. Thiran (2006). Multimodal speaker localization in a probabilistic framework. In *Proc. of European Signal Processing Conference (EUSIPCO)*.

[33] J. Hershey, J. R. Movellan (1999). Audio vision: Using audio-visual synchrony to locate sounds. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, pp. 813–819.

[34] W. Jiang, C. Cotton, S.-F. Chang, D. Ellis, A. C. Loui (2010). Audio-visual atoms for generic video concept classification. *ACM Trans. on Multimedia Computing, Communications, and Applications* **6**(3):1–19.

[35] V. Khalidov, F. Forbes, M. Hansard, E. Arnaud, R. Horaud (2008). Detection and localization of 3D audio-visual objects using unsupervised clustering. In *Int. Conf. on Multimodal Interfaces (ICMI)*, pp. 217–224.

[36] E. Kidron, Y. Y. Schechner, M. Elad (2007). Cross-modal localization via sparsity. *IEEE Trans. on Signal Processing* **55**(4):1390–1404.

[37] V. Kolmogorov, R. Zabih (2002). Multi-camera scene reconstruction via graph cuts. In *Proc. of European Conference on Computer Vision (ECCV)*.

[38] D. Li, N. Dimitrova, M. Li, I. K. Sethi (2003). Multimedia content processing through cross-modal association. In *Proc. of ACM Multimedia (MM '03)*, pp. 604–611.

[39] Y. Li, J. Sun, H.-Y. Shum (2005). Video object cut and paste. *Proc. of ACM SIGGRAPH* **24**(3):595–600.

[40] Y. Liu, Y. Sato (2008). Finding Speaker Face Region by Audiovisual Correlation. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications - M2SFA2 2008*.

[41] Y. Liu, Y. Sato (2009). Visual localization of non-stationary sound sources. In *Proc. of ACM Multimedia (MM '09)*, pp. 513–516.

[42] A. Llagostera Casanovas, G. Monaci, P. Vandergheynst (2007). *Blind Audiovisual Source Separation Using Sparse Redundant Representations*. LTS-REPORT-2007-001, [Online] Available: http://infoscience.epfl.ch/record/99671/files/, EPFL.

[43] A. Llagostera Casanovas, G. Monaci, P. Vandergheynst (2007). Blind Audiovisual Source Separation Using Sparse Representations. In *Proc. of IEEE Int. Conf. Image Processing (ICIP)*, pp. 301–304.

[44] A. Llagostera Casanovas, G. Monaci, P. Vandergheynst, R. Gribonval (2008). Blind Audiovisual Separation based on Redundant Representations. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, pp. 1841–1844.

[45] A. Llagostera Casanovas, G. Monaci, P. Vandergheynst, R. Gribonval (2010). Blind Audio-Visual Source Separation based on Sparse Redundant Representations. *IEEE Trans. on Multimedia* **12**(5):358–371.

[46] A. Llagostera Casanovas, P. Vandergheynst (2010). Audio-based nonlinear video diffusion. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, pp. 2486–2489.

[47] A. Llagostera Casanovas, P. Vandergheynst (2010). Nonlinear Video Diffusion based on Audio-Video Synchrony. *IEEE Trans. on Multimedia* (submitted to), [Online] Available: http://infoscience.epfl.ch/record/151692/ .

[48] A. Llagostera Casanovas, P. Vandergheynst (2011). Unsupervised Extraction of Audio-Visual Objects. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*.

[49] S. Lucey, T. Chen, S. Sridharan, V. Chandran (2005). Integration strategies for audio-visual speech processing: applied to text-dependent speaker recognition. *IEEE Trans. on Multimedia* **7**(3):495–506.

[50] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, P. Suetens (1997). Multimodality image registration by maximization of mutual information. *IEEE Trans. Med. Imag.* **16**(2):187–198.

[51] S. Mallat, Z. Zhang (1993). Matching pursuits with time-frequency dictionaries. *IEEE Trans. on Signal Processing* **41**(12):3397–3415.

[52] H. McGurk, J. W. MacDonald (1976). Hearing lips and seeing voices. *Nature* **264**(5588):746–748.

[53] G. Monaci, O. Divorra, P. Vandergheynst (2006). Analysis of multimodal sequences using geometric video representations. *Signal Processing* **86**(12):3534–3548.

[54] G. Monaci, P. Vandergheynst (2006). Audiovisual gestalts. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition Workshop (CVPRW)*.

[55] G. Monaci et al. (2007). Learning Multi-Modal Dictionaries. *IEEE Trans. on Image Processing* **16**(9):2272–2283.

[56] P. Mrázek (2001). *Nonlinear Diffusion for Image Filtering and Monotonicity Enhancement.* PhD thesis, Czech Technical University, Prague, Czech Republic.

[57] L. Nirenberg (1953). A strong maximum principle for parabolic equations. *Communications on Pure and Applied Mathematics* **6**:167–177.

[58] H. J. Nock, G. Iyengar, C. Neti (2003). Speaker localisation using audio-visual synchrony: An empirical study. In *Proc. of Int. Conf. Image and video retrieval (CIVR)*, vol. 2728, pp. 488–499.

[59] A. Ozerov, C. Fevotte (2010). Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation. *IEEE Trans. on Audio, Speech, and Language Processing* **18**(3):550 –563.

[60] A. Ozerov, P. Philippe, F. Bimbot, R. Gribonval (2007). Adaptation of bayesian models for single channel source separation and its application to voice / music separation in popular music. *IEEE Trans. on Audio, Speech and Language Processing* **15**(5):1564–1578.

[61] Y. C. Pati, R. Rezaiifar, P. S. Krishnaprasad (1993). Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proc. of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers*, pp. 40–44.

[62] E. K. Patterson, S. Gurbuz, Z. Tufekci, J. N. Gowdy (2002). Moving-talker, speaker-independent feature study, and baseline results using the CUAVE multimodal speech corpus. *EURASIP Journal on Applied Signal Processing* **2002**(11):1189.

[63] P. Pérez, J. Vermaak, A. Blake (2004). Data fusion for visual tracking with particles. *Proc. of the IEEE* **92**(3):495–513.

[64] P. Perona, J. Malik (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **12**(7):629–639.

[65] G. Potamianos, C. Neti, G. Gravier, A. Garg, A. W. Senior (2003). Recent advances in the automatic recognition of audiovisual speech. *Proc. of the IEEE* **91**(9):1306–1326.

[66] S. Rajaram, A. V. Nefian, T. Huang (2004). Bayesian separation of audio-visual speech sources. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 5, pp. 657–660.

[67] B. Rivet, L. Girin, C. Jutten (2007). Mixing audiovisual speech processing and blind source separation for the extraction of speech signals from convolutive mixtures. *IEEE Trans. on Audio, Speech and Language Processing* **15**(1):96–108.

[68] C. Rother, V. Kolmogorov, A. Blake (2004). Grabcut: Interactive foreground extraction using iterated graph cuts. *Proc. of ACM SIGGRAPH* **23**(3):309–314.

[69] C. Saraceno, R. Leonardi (1999). Indexing audiovisual databases through joint audio and video processing. *International Journal of Imaging Systems and Technology* **9**(5):320–331.

[70] D. Scharstein, R. Szeliski (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision* **47**(1/2/3):7–42.

[71] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, R. Szeliski (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 519–528.

[72] S. M. Seitz, C. R. Dyer (1999). Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision* **35**(2):151–173.

[73] C. Sigg, B. Fischer, B. Ommer, V. Roth, J. Buhmann (2007). Nonnegative cca for audiovisual source separation. In *IEEE Workshop on Machine Learning for Signal Processing*.

[74] M. Siracusa, J. Fisher (2007). Dynamic dependency tests: Analysis and applications to multimodal data association. In *Int. Conf. on Artificial Intelligence and Statistics (AIStats)*.

[75] M. Slaney, M. Covell (2000). Facesync: A linear operator for measuring synchronization of video facial images and audio tracks. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, pp. 814–820.

[76] P. Smaragdis, M. Casey (2003). Audio/visual independent components. *Proc. of Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA)* pp. 709–714.

[77] D. Sodoyer, L. Girin, C. Jutten, J.-L. Schwartz (2004). Developing an audio-visual speech source separation algorithm. *Speech Commununication* **44**(1-4):113–125.

[78] W. H. Sumby, I. Pollack (1954). Visual contribution to speech intelligibility in noise. *Journal of the Acoustical Society of America* **26**(2):212–215.

[79] Q. Summerfield (1987). Some preliminaries to a comprehensive account of audio-visual speech perception. In B. Dodd, R. Campbell (eds.), *Hearing by Eye: The Psychology of Lipreading*, pp. 3–51, Lawrence Erlbaum Associates.

[80] E. Vincent, C. Fevotte, R. Gribonval (2006). Performance measurement in blind audio source separation. *IEEE Trans. on Audio, Speech and Language Processing* **14**(4):1462–1469.

[81] E. Vincent, R. Gribonval, M. D. Plumbley (2007). Oracle estimators for the benchmarking of source separation algorithms. *Signal Processing* **87**(8):1933–1950.

[82] J. Vroomen, M. Keetels (2010). Perception of intersensory synchrony: A tutorial review. *Attention, Perception & Psychophysics* **72**(4):871–884.

[83] J. Wang, M. F. Cohen (2007). Image and video matting: a survey. *Foundation and Trends in Computer Graphics and Vision* **3**(2):97–175.

[84] W. Wang, D. Cosker, Y. Hicks, S. Saneit, J. Chambers (2005). Video assisted speech source separation. In *Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 5, pp. 425–428.

[85] J. Weickert (1998). *Anisotropic Diffusion in Image Processing*. Teubner, Stuttgart, Germany.

[86] M. Wimmer, B. Schuller, D. Arsic, B. Radig, G. Rigoll (2008). Low-level fusion of audio and video feature for multi-modal emotion recognition. In *Int. Conf. Computer Vision Theory and Applications (VISAPP)*, vol. 2, pp. 145–151.

[87] O. Yilmaz, S. Rickard (2004). Blind separation of speech mixtures via time-frequency masking. *IEEE Trans. on Signal Processing* **52**(7):1830–1847.

Index

Curriculum vitæ

Name:	Anna Llagostera Casanovas
Citizenship:	Spanish
Birthdate:	September 18, 1982
Birthplace:	Valls, Spain
Marital status:	Single

Contact information

Address:	Av. du Denantou 21 1006 Lausanne, Switzerland
Phone:	+41 21 693 36 32
Fax:	+41 21 693 76 00
Email:	anna.llagostera@epfl.ch
Web page:	http://lts2www.epfl.ch/∼llagoste

Work experience

- ***2006 – present***: Research assistant at the Signal Processing Laboratory, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland
  - PhD Thesis research in the field of joint audio-visual signal processing.
  - Teaching: supervision of master thesis and teaching assistant for the Signals and Systems course.
- ***May 2007, 2008***: Visiting Researcher at the Centre de Recherche IRISA-INRIA, Rennes, France
  - Integration of audio-visual processing in a sound source separation framework.
- ***Summer 2003, 2004, 2005***: Software Engineer at LEAR corporation, Valls, Spain
  - Automotive Industry, Electronic Systems Division.
  - Validation of the Smart Junction Boxes software, which is designed by the programmers in order to match the manufacturer specifications.

Education

- ***2006 – present***: *Ph. D. student* in signal processing. Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.
- ***2006***: *Msc Thesis* "Blind Audio-Visual Source Separation using Sparse Redundant Representations" at the Signal Processing Laboratory, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.
- ***2000 – 2006***: *Msc in Telecommunications Engineering.* Technical University of Catalonia (UPC), Spain.
  - Communication systems, signal processing, computer programming, wireless communications, networking and computer networks, antenna design, electronics, operating systems, economics and business management.
  - Class rank: 39 out of 268 graduated students
- ***2004 – 2006***: *Msc European Masters in Language and Speech.* Technical University of Catalonia (UPC), Spain.
  - Speech signal processing, pattern recognition, natural language processing, language engineering applications, theoretical linguistics, phonetics, phonology.

Skills

Languages

Catalan:	mother tongue
Spanish:	mother tongue
English:	fluent oral and written
French:	fluent oral and written

Computer skills

Operating systems:	Mac OS X, Windows, Linux
Programming languages:	MATLAB, C, C++, Visual Basic, Java, Python
Office tools:	Microsoft Word, Excel and PowerPoint
Web Design:	HTML, PHP, MySQL, Dreamweaver
Other Tools:	LaTeX, LabVIEW, CAD

Complementary aspects

- Car driving license

Personal publications

Journal papers

- A. Llagostera Casanovas and P. Vandergheynst *Nonlinear Video Diffusion based on Audio-Video Synchrony*, submitted to IEEE Trans. on Multimedia, 2010.
- A. Llagostera Casanovas, G. Monaci and P. Vandergheynst, *Blind Audio-Visual Source Separation based on Sparse Redundant Representations*, IEEE Trans. on Multimedia, Vol. 12, Nr. 5, pp. 358-371, 2010.

Conference papers

- A. Llagostera Casanovas and P. Vandergheynst, *Unsupervised Extraction of Audio-Visual Objects*, accepted to IEEE Int. Conf. on Acoustic, Speech, Signal Processing, 2011.
- A. Llagostera Casanovas and P. Vandergheynst, *Audio-based nonlinear video diffusion*, Proc. of IEEE Int. Conf. on Acoustic, Speech, Signal Processing, pp. 2486 - 2489, 2010.
- A. Llagostera Casanovas, G. Monaci, P. Vandergheynst, *Blind Audiovisual Separation based on Redundant Representations*, Proc. of IEEE Int. Conf. on Acoustic, Speech, Signal Processing, pp. 1841-1844, 2008.
- A. Llagostera Casanovas, G. Monaci, P. Vandergheynst, *Blind Audiovisual Source Separation Using Sparse Representations*, Proc. of IEEE Int. Conf. on Image Processing, pp. 301-304, 2007.

Technical reports

- A. Llagostera Casanovas, G. Monaci, P. Vandergheynst, *Blind Audiovisual Source Separation Using Sparse Redundant Representations*, Technical Report LTS-REPORT-2007-001, 2007.